AUTHORIZED DOCUMENTATION

implementation Guide

Novell®
Identity Manager Driver for Active Directory*

3.5.1

September 28, 2007

www.novell.com

Legal Notices

Novell, Inc. makes no representations or warranties with respect to the contents or use of this documentation, and specifically disclaims any express or implied warranties of merchantability or fitness for any particular purpose. Further, Novell, Inc. reserves the right to revise this publication and to make changes to its content, at any time, without obligation to notify any person or entity of such revisions or changes.

Further, Novell, Inc. makes no representations or warranties with respect to any software, and specifically disclaims any express or implied warranties of merchantability or fitness for any particular purpose. Further, Novell, Inc. reserves the right to make changes to any and all parts of Novell software, at any time, without any obligation to notify any person or entity of such changes.

Any products or technical information provided under this Agreement may be subject to U.S. export controls and the trade laws of other countries. You agree to comply with all export control regulations and to obtain any required licenses or classification to export, re-export or import deliverables. You agree not to export or re-export to entities on the current U.S. export exclusion lists or to any embargoed or terrorist countries as specified in the U.S. export laws. You agree to not use deliverables for prohibited nuclear, missile, or chemical biological weaponry end uses. See the Novell International Trade Services Web page (http://www.novell.com/info/exports/) for more information on exporting Novell software. Novell assumes no responsibility for your failure to obtain any necessary export approvals.

Novell, Inc. has intellectual property rights relating to technology embodied in the product that is described in this document. In particular, and without limitation, these intellectual property rights may include one or more of the U.S. patents listed on the Novell Legal Patents Web page (http://www.novell.com/company/legal/patents/) and one or more additional patents or pending patent applications in the U.S. and in other countries.

Novell, Inc.
404 Wyman Street, Suite 500
Waltham, MA 02451
U.S.A.
www.novell.com

Online Documentation: To access the latest online documentation for this and other Novell products, see the Novell Documentation Web page (http://www.novell.com/documentation).

Novell Trademarks

Third-Party Materials

Contents

About This Guide

This guide explains how to install, configure, and manage the Identity Manager Driver for Active Directory.

Audience

This guide is intended for Active Directory administrators, Novell® eDirectory™ administrators, and others who implement the Identity Manager driver for NT Domains.

Feedback

We want to hear your comments and suggestions about this manual and the other documentation included with this product. Please use the User Comments feature at the bottom of each page of the online documentation, or go to www.novell.com/documentation/feedback.html and enter your comments there.

Documentation Updates

For the most recent version of this document, see *Identity Manager Driver for Active Directory* in the Identity Manager Drivers section on the Novell Documentation Web site (http://www.novell.com/documentation/idm35drivers/index.html).

Additional Documentation

For documentation on using Identity Manager and the other Identity Manager drivers, see the Identity Manager Documentation Web site (http://www.novell.com/documentation/idm35).

Documentation Conventions

In Novell documentation, a greater-than symbol (>) is used to separate actions within a step and items in a cross-reference path.

A trademark symbol (®, ™, etc.) denotes a Novell trademark. An asterisk (*) denotes a third-party trademark.

Overview 1

This section contains high-level information about how the Active Directory driver functions.

- Section 1.1, "Key Terms," on page 11
- Section 1.2, "What's New," on page 13
- Section 1.3, "Data Transfers Between Systems," on page 13
- Section 1.4, "Key Driver Features," on page 13
- Section 1.5, "Default Driver Configuration," on page 14

1.1 Key Terms

- Section 1.1.1, "Identity Manager," on page 11
- Section 1.1.2, "Connected System," on page 11
- Section 1.1.3, "Identity Vault," on page 11
- Section 1.1.4, "Metadirectory Engine," on page 12
- Section 1.1.5, "Active Directory Driver," on page 12
- Section 1.1.6, "Driver Shim," on page 12
- Section 1.1.7, "Remote Loader," on page 12

1.1.1 Identity Manager

Novell® Identity Manager is a service that synchronizes data among servers in a set of connected systems by using a robust set of configurable policies. Identity Manager uses the Identity Vault to store shared information, and uses the Metadirectory engine for policy-based management of the information as it changes in the vault or connected system. Identity Manager runs on the server where the Identity Vault and the Metadirectory engine are located.

1.1.2 Connected System

A connected system is any system that can share data with Identity Manager through a driver. Active Directory is a connected system.

1.1.3 Identity Vault

The Identity Vault is a persistent database powered by eDirectory™ and used by Identity Manager to hold data for synchronization with a connected system. The vault can be viewed narrowly as a private data store for Identity Manager or more broadly as a metadirectory that holds enterprise-wide data. Data in the vault is available to any protocol supported by eDirectory, including NCP™ (the traditional protocol used by such utilities as ConsoleOne® and iManager), LDAP, and DSML.

Because the vault is powered by eDirectory, Identity Manager can be easily integrated into your corporate directory infrastructure by using your existing directory tree as the vault.

1.1.4 Metadirectory Engine

The Metadirectory engine is the core server that implements the event management and policies of Identity Manager. The engine runs on the Java* Virtual Machine in eDirectory.

1.1.5 Active Directory Driver

A driver implements data sharing policy for a connected system. You control the actions of the driver by using iManager to define the filters and policy. For Active Directory, a driver implements policy for a single domain.

1.1.6 Driver Shim

A driver shim is the component of a driver that converts the XML-based Identity Manager command and event language (XDS) to the protocols and API calls needed to interact with a connected system. The shim is called to execute commands on the connected system after the Output Transform has been run. Commands are usually generated on the Subscriber channel but can be generated by command write-back on the Publisher channel.

The shim also generates events from the connected system for the Input Transformation policy. A driver shim can be implemented either in Java class or as a native Windows* DLL file. The shim for Active Directory is `ADDriver.dll`.

`ADDriver.dll` is implemented as a native Windows DLL file. ADDriver uses several different Windows APIs to integrate with Active Directory. These APIs typically require some type of login and authentication to succeed. Also, the APIs might require that the login account have certain rights and privileges within Active Directory and on the machine where ADDriver.dll executes.

If you use the Remote Loader, `ADDriver.dll` executes on the server where the Remote Loader is running. Otherwise, it executes on the server where the Metadirectory engine is running.

1.1.7 Remote Loader

A Remote Loader enables a driver shim to execute outside of the Metadirectory engine (perhaps remotely on a different machine). The Remote Loader is typically used when a requirement of the driver shim is not met by the Identity Manager server. For example, if the Metadirectory engine is running on Linux*, the Remote Loader is used to execute the Active Directory driver shim on a Windows server.

The Remote Loader is a service that executes the driver shim and passes information between the shim and the Metadirectory engine. When you use a Remote Loader, you install the driver shim on the server where the Remote Loader is running, not on the server where the Metadirectory engine is running. You can choose to use SSL to encrypt the connection between the Metadirectory engine and the Remote Loader. For more information, see "Deciding Whether to Use the Remote Loader" in the *Novell Identity Manager 3.5.1 Administration Guide*.

When you use the Remote Loader with the Active Directory driver shim, two network connections exist:

- Between the domain controller and the Remote Loader
- Between Active Directory and the Active Directory driver shim

1.2 What's New

- Section 1.2.1, "New Driver Features," on page 13
- Section 1.2.2, "New Identity Manager Features," on page 13

1.2.1 New Driver Features

- Supports synchronization with Exchange 2007 accounts. For more information, see Appendix C, "Provisioning Exchange Accounts," on page 147.
- Supports synchronization with ADAM instance. For more information, see Appendix B, "Configuring for Use with an ADAM Instance," on page 137.
- Password synchronization can be disabled on an Active Directory driver. See Section 7.8, "Disabling Password Synchronization on a Driver," on page 87 for more information.
- There was formerly a limitation with synchronizing large groups. Microsoft* has limitation of 1,000 user for Windows 2000 servers and a limitation of 1,500 users for Windows 2003 servers. The driver now is able to break these barriers. It can synchronize large groups without causing problems.
- 64-bit support for the password synchronization filter.

1.2.2 New Identity Manager Features

For information about the new features in Identity Manager, see "What's New in Identity Manager 3.5.1? " in the *Identity Manager 3.5.1 Installation Guide*.

1.3 Data Transfers Between Systems

This sections explains how the data flows between Active Directory and the Identity Vault.

The Active Directory driver supports Publisher and Subscriber channels.

The Publisher channel does the following:

- Reads events from Active Directory for the domain hosted on the server that the driver shim is connecting to.
- Submits that information to the Identity Vault.

The Subscriber channel does the following:

- Watches for additions and modifications to the Identity Vault objects.
- Makes changes to Active Directory that reflect those changes.

You can configure the driver so that both Active Directory and the Identity Vault are allowed to update a specific attribute. In this configuration, the most recent change determines the attribute value, except in the case of merge operations that are controlled by the filters and merge authority.

1.4 Key Driver Features

The sections below contains a list of the key driver features.

- Section 1.4.1, "Local Platforms," on page 14

- Section 1.4.2, “Remote Platforms,” on page 14
- Section 1.4.3, “Role-Based Entitlements,” on page 14
- Section 1.4.4, “Password Synchronization Support,” on page 14
- Section 1.4.5, “Information Synchronized,” on page 14

1.4.1 Local Platforms

The Active Directory driver can be installed locally on the following platforms:

- Windows 2000, or 2003 with the latest Service Patch

1.4.2 Remote Platforms

The Active Directory driver can use the Remote Loader service. The Remote Loader service for the Active Directory driver can be installed on Windows 2000, or 2003 with the latest Service Patch

For more information about installing the Remote Loader services, see “Installing the Remote Loader” in the *Novell Identity Manager 3.5.1 Administration Guide*.

1.4.3 Role-Based Entitlements

The Active Directory driver does support Role-Based Entitlements, if it is selected during the configuration of the driver. The Role-Based Entitlement is revoked, the account is either disabled or deleted. For more information, see “When account entitlement revoked” on page 174 in Appendix E, “Properties of the Driver,” on page 167.

1.4.4 Password Synchronization Support

The Active Directory driver synchronizes passwords on both the Subscriber channel and the Publisher channel. For more information, see Chapter 7, “Password Synchronization,” on page 65.

1.4.5 Information Synchronized

The Active Directory driver synchronizes User objects, Group objects, containers, and Exchange mailboxes. Exchange mailboxes are only synchronized if the options to manage Exchange mailboxes is selected during the configuration of the driver. For more information about the driver configuration, see Section 1.5, “Default Driver Configuration,” on page 14. For more information about synchronizing Exchange accounts, see Section 5.4, “Default Configuration Parameters,” on page 52.

1.5 Default Driver Configuration

The Active Directory driver is shipped with a default configuration file called `ActiveDirectory-IDM3_5_0-V1.xml`. When imported with Designer or iManager, this configuration file creates a driver with a set of rules and policies suitable for synchronizing with Active Directory. If your requirements for the driver are different from the default policies, you need to change them to carry out the policies you want. Pay close attention to the default Matching policies. The data that you trust to match users usually is different from the default. The policies themselves are commented and you can gain a greater understanding of what they do by importing a test driver and reviewing the policies with Designer or iManager.

1.5.1 User Object Name Mapping

Management utilities for the Identity Vault such as iManager and ConsoleOne typically name user objects differently than the Users and Computers snap-in for the Microsoft Management Console (MMC). Make sure that you understand the differences so the Matching policy and any Transformation policies you have are implemented properly.

1.5.2 Data Flow

Data can flow between Active Directory and an Identity Vault. The flow of data is controlled by the policies that are in place for the Active Directory driver.

Policies

Policies control data synchronization between Active Directory and an Identity Vault.

During the driver configuration, the Active Directory configuration file enables you to select several options that affect the default policies and filters created for you. Table 1-1 lists these options and how they affect policies and filters that are created:

Table 1-1 *Data Flow Options*

Option	Description
	Configure Data Flow establishes the initial driver filter that controls the classes and attributes that will be synchronized. The purpose of this option is to configure the driver to best express your general data flow policy. It can be changed after import to reflect specific requirements. *Bidirectional* sets classes and attributes to synchronize on both the Publisher and Subscriber channels. A change in either the Identity Vault or Active Directory is reflected on the other side. Use this option if you want both sides to be authoritative sources of data. *AD to Vault* sets class and attributes to synchronize on the Publisher channel only. A change in Active Directory is reflected in the Identity Vault but Identity Vault changes are ignored. Use this option if you want Active Directory to be the authoritative source of data. *Vault to AD* sets classes and attributes to synchronize on the Subscriber channel only. A change in the Identity Vault is reflected in Active Directory but Active Directory changes are ignored. Use this option if you want the vault to be the authoritative source of data.

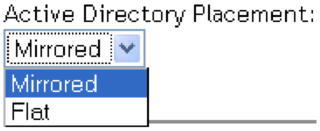
Option	Description
	Publisher Placement controls where objects are created in the Identity Vault. *Mirrored* places objects in the Identity Vault in the same hierarchy as they exist in Active Directory. *Flat* places all objects in the base container in the Identity Vault specified during configuration.
	Subscriber Placement controls how objects are placed in Active Directory. *Mirrored* places objects in Active Directory in the same hierarchy as they exist in the Identity Vault. *Flat* places all objects in the base container in Active Directory specified during configuration.

Table 1-2 lists default policies and describes how selections during configuration affect the polices:

Table 1-2 *Default Policies*

Policy	Description
Create	In either the mirrored or flat hierarchy, you must define Full Name to create an Active Directory user as a user in the Identity Vault.
Matching	In a mirrored hierarchy, the Matching policy attempts to match an object in the same position in the hierarchy. In a flat hierarchy, the Matching policy attempts to match the user with an object that has the same Full Name in the base container that you specify.
Placement	In a mirrored hierarchy, the Placement policy places all objects in a hierarchy that mirrors the hierarchy of the data store sending the operation. In a flat hierarchy, the Placement policy places all objects in the base container that you specify.

Schema Mapping

Table 1-3 in this section list Identity Vault user, group, and Organizational Unit attributes that are mapped to Active Directory user and group attributes.

The mappings listed in the tables are default mappings. You can remap same-type attributes.

Table 1-3 *Mapped User Attributes*

eDirectory - User	Active Directory - user
DirXML-ADAliasName	sAMAccountName
L	PhysicalDeliveryOfficeName
Physical Delivery Office Name	l

eDirectory - User	Active Directory - user
nspmDistributionPassword	nspmDistributionPassword

Table 1-4 Mapped Group Attributes

eDirectory - Group	Active Directory - group
DirXML-ADAliasName	sAMAccountName

eDirectory's L attribute is mapped to Active Directory's physicalDeliveryOfficeName attribute, and eDirectory's Physical Delivery Office Name attribute is mapped to Active Directory's L attribute. Because similarly named fields have the same value, mapping the attributes this way enables the attributes to work well with ConsoleOne and the Microsoft Management Console.

Table 1-5 Mapped Organizational Unit Attributes

eDirectory - Organizational Unit	Active Directory - organizationalUnit
L	physicalDeliveryOfficeName
Physical Delivery Office Name	l

Table 1-6 Mapped Organization Attributes

eDirectory - Organization	Active Directory - organization
L	physicalDeliveryOfficeName
Physical Delivery Office Name	l

The driver maps the Locality class, but there are no attributes for the class.

Table 1-7 Mapped Locality Class

eDirectory	Active Directory
Locality	locality

Table 1-8 Mapped Non-Class Specific Attributes

eDirectory	Active Directory
CN	cn
Description	description
DirXML-EntitlementRef	DirXML-EntitlementRef
DirXML-EntitlementResult	DirXML-EntitlementResult

eDirectory	Active Directory
Facsimile Telephone Number	facsimiletelephoneNumber
Full name	displayName
Given Name	givenName
Group Membership	memberOf
Initials	initials
Internet EMail Address	mail
Login Allowed Time Map	logonHours
Login Disabled	dirxml-uACAccountDisabled
Login Expiration Time	accountExpires
Login Intruder Reset Time	lockoutTime
Member	member
OU	ou
Owner	managedBy
Postal Code	PostalCode
Postal Office Box	postOfficeBox
S	st
SA	streetAddress
See Also	seeAlso
DirXML-SPEntitlements	DirXML-SPEntitlements
Surname	sn
Telephone Number	telephoneNumber
Title	title

Name Mapping Policies

The default configuration includes two name mapping policies that work together to help you reconcile different naming policies between the Identity Vault and Active Directory. When you create a user with the Active Directory Users and Computers tool (a snap-in for the Microsoft Management Console and abbreviated as MMC in this document) you see that the user full name is used as its object name. Attributes of the user object define Pre-Windows 2000 Logon Name (also known as the NT Logon Name or sAMAccountName) and the Windows 2000 Logon Name (also known as the userPrincipalName). When you create a user in the Identity Vault with iManager or ConsoleOne, the object name and the user logon name are the same.

If you create some users in Active Directory using MMC and other objects in the Identity Vault or another connected system that is synchronized with the Identity Vault, the object can look odd in the opposing console and might fail to be created in the opposing system at all.

The Full Name Mapping Policy is used to manage objects in Active Directory using the MMC conventions. When enabled, The Full Name attribute in the Identity Vault is synchronized with the object name in Active Directory.

The NT Logon Name Mapping Policy is used to manage objects in Active Directory using the Identity Vault conventions. When enabled, the Identity Vault object name is used to synchronize both the object name and NT Logon Name in Active Directory. Objects in Active Directory have the same names as the Identity Vault and the NT Logon Name matches the Identity Vault logon name.

When both of the policies are enabled at the same time, the Active Directory object name are the Identity Vault Full Name, but the NT Logon Name matches the Identity Vault logon name.

When both policies are disabled, no special mapping is made. The object names is synchronized and there will be no special rules for creating the NT Logon Name. Because the NT Logon Name is a mandatory attribute in Active Directory, you need some method of generating it during Add operations. The NT Logon Name (sAMAccountName) is mapped to the DirMXL-ADAliasName in the Identity Vault, so you could either use that attribute to control the NT Logon Name in Active Directory or build your own policy in the Subscriber Create policies to generate one. With this policy selection, users created with MMC use the object name generated by MMC as the object name in the Identity Vault. This name might be inconvenient for login to the Vault.

Windows 2000 Logon Name Policies

The Windows 2000 Logon name (also known as the userPrincipalName or UPN) does not have a direct counterpart in the Identity Vault. UPN looks like an e-mail address (user@mycompany.com) and might in fact be the user’s e-mail name. The important thing to remember when working with UPN is that it must use a domain name (the part after the @ sign) that is configured for your domain. You can find out what domain names are allowed by using MMC to create a user and inspecting the domain name drop-down box when adding the UPN.

The default configuration offers several choices for managing userPrincipalName. If your domain is set up so that the user’s e-mail address can be used as a userPrincipalName, then one of the options to track the user’s e-mail address is appropriate. You can make userPrincipalName follow either the Identity Vault or Active Directory e-mail address, depending on which side is authoritative for e-mail. If the user e-mail address is not appropriate, you can choose to have a userPrincipalName constructed from the user logon name plus a domain name. If more than one name can be used, update the policy after import to make the selection. If none of these options are appropriate, then you can disable the default policies and write your own.

Entitlements

Entitlements make it easier to integrate Identity Manager with the Identity Manager User Application and Role-Based Services in eDirectory. When using the User Application, an action such as provisioning an account in Active Directory is delayed until the proper approvals have been made. When using Role-Based Services, rights assignments are made based on attributes of a user object and not by regular group membership. Both of these services offer a challenge to Identity Manager because it is not obvious from the attributes of an object whether an approval has been granted or the user matches a role.

Entitlements standardize a method of recording this information on objects in the Identity Vault. From the driver perspective, an entitlement grants or revokes the right to something in Active Directory. You can use entitlements to grant the right to an account in Active Directory, to control group membership, and to provision Exchange mailboxes. The driver is unaware of the User Application or Role-Based Entitlements. It depends on the User Application server or the Entitlements driver to grant or revoke the entitlement for a user based upon its own rules.

You should enable entitlements for the driver only if you plan to use the User Application or Role-Based Entitlements with the driver.

Preparing Active Directory

2

In this section:

- Section 2.1, "Active Directory Prerequisites," on page 21
- Section 2.2, "Where to Install the Active Directory Driver," on page 21
- Section 2.3, "Addressing Security Issues," on page 23
- Section 2.4, "Creating an Administrative Account," on page 28
- Section 2.5, "Becoming Familiar with Driver Features," on page 28

2.1 Active Directory Prerequisites

- ☐ Novell® Identity Manager 3.5 and its prerequisites, as listed in the "Installing Identity Manager" section in the *Identity Manager 3.5.1 Installation Guide*.
- ☐ Windows 2003 Server, or Windows 2000 Server with Service Pack 2 or later.
- ☐ Internet Explorer 5.5 or later on the server running the Active Directory (AD) driver and on the target domain controller.
- ☐ Active Directory domain controller DNS name or IP address, depending on the authentication method.

Also, we recommend that the server hosting the Active Directory driver be a member of the Active Directory domain. This is required to provision Exchange mailboxes and synchronize passwords. If you don't require these features, the server can be a member of any domain as long as the Simple (simple bind) authentication mode is used. To have bidirectional password synchronization, the Negotiate authentication option must be selected.

If you want to synchronize with an ADAM instance, see Appendix B, "Configuring for Use with an ADAM Instance," on page 137 for more information.

If you want to synchronize Exchange accounts, see Appendix C, "Provisioning Exchange Accounts," on page 147.

2.2 Where to Install the Active Directory Driver

The Active Directory driver shim must run on one of the supported Windows platforms. However, you don't need to install the Metadirectory engine on this same machine. Using a Remote Loader, you can separate the engine and the driver shim, allowing you to balance the load on different machines or accommodate corporate directives.

The installation scenario you select determines how the driver shim is installed. If you choose to install the driver shim on the same machine as Identity Manager (where the Metadirectory engine and the Identity Vault are located), Identity Manager calls the driver shim directly. If you choose to install the driver shim on another machine, you must use the Remote Loader.

The driver itself is installed the same way in each of the scenarios. See Chapter 5, "Configuring the Active Directory Driver," on page 49.

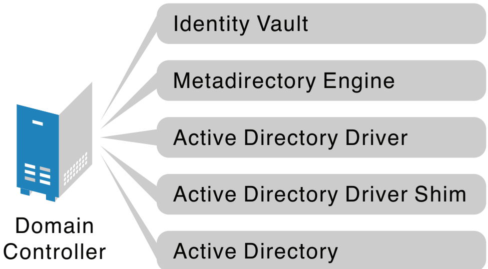
You can install the Active Directory driver on either the domain controller or a member server. Before you start the driver installation, determine where you want to install the driver.

- Section 2.2.1, "Local Installation," on page 22
- Section 2.2.2, "Remote Installation on Windows Server Only," on page 22
- Section 2.2.3, "Remote Installation on Windows and Other Platforms," on page 23
- Section 2.2.4, "Remote Installation on a Windows Member Server," on page 23

2.2.1 Local Installation

A single Windows domain controller can host the Identity Vault, the Metadirectory engine, and the driver.

Figure 2-1 *Scenario 1 - All Components Are on One Server*



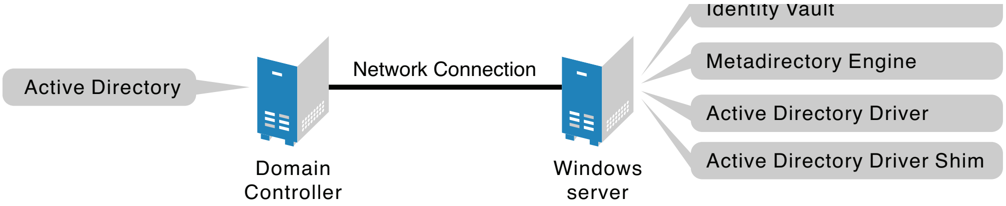
This configuration works well for organizations that want to save on hardware costs. It is also the highest-performance configuration because there is no network traffic between Identity Manager and Active Directory.

However, hosting Identity Vault and the Metadirectory engine on the domain controller increases the overall load on the controller and increases the risk that the controller might fail. Because domain controllers play a critical role in Microsoft networking, many organizations are more concerned about the speed of the domain authentication and the risks associated with a failure on the domain controller than about the cost of additional hardware.

2.2.2 Remote Installation on Windows Server Only

You can install the Identity Vault, the Metadirectory engine, and the driver on a separate computer from the Active Directory domain controller. This configuration leaves the domain controller free of any Identity Manager software.

Figure 2-2 *Scenario 2 - Active Directory and the Driver Shim on Separate Servers*

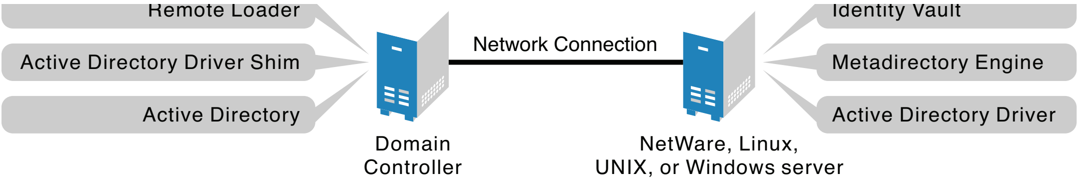


This configuration is attractive if corporate policy disallows running the driver on your domain controller.

2.2.3 Remote Installation on Windows and Other Platforms

You can install the Remote Loader and driver shim on the Active Directory domain controller, but install the Identity Vault and the Metadirectory engine on a separate server.

Figure 2-3 *Scenario 3 - Active Directory, the Remote Loader, and Driver Shim on One Server*



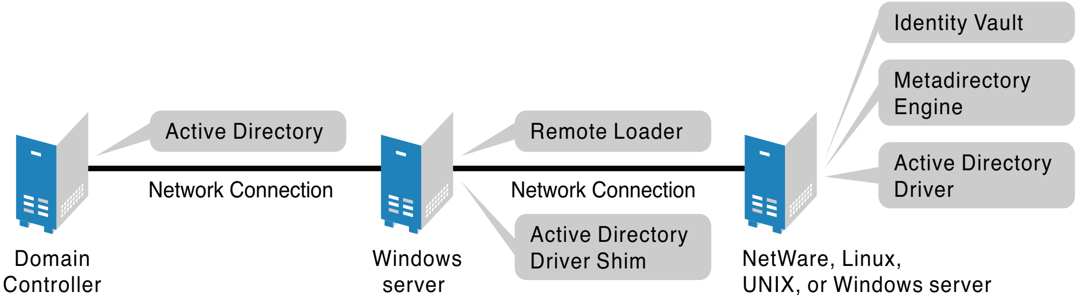
This configuration is attractive if your Identity Vault and Metadirectory engine (Identity Manager) installations are on a platform other than one of the supported versions of Windows.

Both Scenario 2 and Scenario 3 configurations eliminate the performance impact of hosting the Identity Vault and the Metadirectory engine on the domain controller.

2.2.4 Remote Installation on a Windows Member Server

If you have platform requirements and domain controller restrictions in place, you can use a three-server configuration.

Figure 2-4 *Scenario 4 - Three-server Configuration*



This configuration is more complicated to set up, but it accommodates the constraints of some organizations. In this figure, the two Windows servers are member servers of the domain.

2.3 Addressing Security Issues

The major security issues to consider are authentication, encryption, and use of the Remote Loader. If you have Windows 2003 or Windows 2000 SP3 or later, consider a security option called signing. See "Digitally sign communications" on page 132 in Chapter 12, "Security: Best Practices," on page 131.

A simple prescription for managing security is not possible because the security profile available from Windows varies with the service pack, DNS server infrastructure, domain policy, and local policy settings on the server. The following sections explain your security choices and provide suggested configurations. When implementing your driver and when upgrading components, pay close attention to security.

- Section 2.3.1, “Authentication Methods,” on page 24
- Section 2.3.2, “Encryption,” on page 24
- Section 2.3.3, “SSL Connection Between the Remote Loader and Identity Manager,” on page 27

2.3.1 Authentication Methods

Authentication identifies the driver shim to Active Directory and, potentially, the local machine. To authenticate to Active Directory, you can use either the Negotiate method or the Simple (simple bind) method.

Table 2-1 *Authentication Methods*

Authentication Method	Description	Advantages	Disadvantages
Negotiate	The preferred method. Uses Kerberos*, NTLM, or a pluggable authentication scheme if one is installed.	The driver can be installed on any server in the domain.	The server hosting the driver must be a member of the domain.
Simple	Used when the server hosting the driver shim is not a member of the domain.	The driver can be installed on a server that is not a member of the domain.	Some provisioning services are unavailable, such as Exchange mailbox provisioning and password synchronization.

2.3.2 Encryption

SSL encrypts data. Depending on your configuration, SSL can be used in two places:

- Between the Active Directory driver and the domain controller
- Between the Identity Vault and the Remote Loader running the Active Directory driver

Password synchronization occurs between Active Directory and the Identity Vault. You need to make sure that you use SSL with any communication that goes across the network.

If the Metadirectory engine, Identity Vault, the Active Directory driver, and Active Directory are on the same machine, you don’t need SSL. Communication isn’t going across the network.

However, if you are accessing Active Directory remotely by using an Active Directory driver shim on a member server, you need to set up SSL between the Active Directory driver shim and Active Directory. You do this by setting the SSL parameter to *Yes* on the driver configuration. See Step 5 on page 27, in “SSL Connection Between the Active Directory Driver and the Domain Controller” on page 25.

If you are using the Remote Loader on the Domain Controller, you can set up SSL between the Metadirectory engine and the Remote Loader. For additional information on SSL and Remote Loaders, see "Deciding Whether to Use the Remote Loader" in the *Novell Identity Manager 3.5.1 Administration Guide*.

The following table outlines where SSL connections can be used for each of the scenarios discussed in Section 2.2, "Where to Install the Active Directory Driver," on page 21:

Table 2-2 *SSL Connects*

Configuration	SSL Connections Available
Single-Server	No SSL connections are necessary.
Two-Server: Identity Manager and the Active Directory driver are on the same server	An SSL connection can be established between the Active Directory driver and the domain controller.
Dual-Server: Identity Manager is on one server but the Active Directory driver is on a separate server	An SSL connection can be established between Identity Manager and the Remote Loader running the Active Directory driver.
Three-Server	An SSL connection can be established between the Active Directory driver and the domain controller. An SSL connection can also be established between Identity Manager and the Remote Loader running the Active Directory driver.

SSL Connection Between the Active Directory Driver and the Domain Controller

To make SSL connections to an Active Directory domain controller, you must be set up to use SSL. This involves setting up a certificate authority, then creating, exporting, and importing the necessary certificates.

Setting Up a Certificate Authority

Most organizations already have a certificate authority. If this is the case for your organization, you need to export a valid certificate, then import it to the certificate store on your domain controller. The server hosting the driver shim must trust the root certificate authority that the issuing certificate authority of this certificate chains to.

If you do not have a certificate authority in your organization, you must establish one. Novell, Microsoft, and several other third parties provide the tools necessary to do this. Establishing a certificate authority is beyond the scope of this guide. For more information, see

- *Novell Certificate Server™ 2.5 Administration Guide* (http://www.novell.com/documentation/lg/crt252/index.html)
- *Securing Windows 2000 Server* (http://www.microsoft.com/technet/security/prodtech/windows2000/secwin2k/swin2kad.mspx)

Creating, Exporting, and Importing Certificates

After you have a certificate authority, the LDAP server must have the appropriate server authentication certificate installed, for LDAP SSL to operate successfully. Also, the server hosting the driver shim must trust the authority that issued those certificates. Both the server and the client must support 128-bit encryption.

1 Generate a certificate that meets the following Active Directory LDAP service requirements:

- The LDAPS certificate is located in the Local Computer's Personal certificate store (programmatically known as the computer's MY certificate store).
- A private key matching the certificate is present in the Local Computer's store and is correctly associated with the certificate.

  The private key must not have strong private-key protection enabled.
- The Enhanced Key Usage extension includes the Server Authentication (1.3.6.1.5.5.7.3.1) object identifier (also known as OID).
- The Active Directory fully qualified domain name (for example, DC01.DOMAIN.COM) of the domain controller appears in one of the following places:
  - The Common Name (CN) in the Subject field.
  - The DNS entry in the Subject Alternative Name extension.
- The certificate was issued by a CA that the domain controller and the LDAPS clients trust.

  Trust is established by configuring the clients and the server to trust the root CA that the issuing CA chains to.

This certificate permits the LDAP service on the domain controller to listen for and automatically accept SSL connections for both LDAP and global catalog traffic.

NOTE: This information appears in the Microsoft Knowledge Base Article 321051, How to Enable LDAP over SSL with a Third-Party Certificate Authority (http://support.microsoft.com/default.aspx?scid=kb;en-us;321051). Consult this document for the latest requirements and additional information.

2 Export this certificate in one of the following standard certificate file formats supported by Windows 2000:

- Personal Information Exchange (PFX, also called PKCS #12)
- Cryptographic Message Syntax Standard (PKCS #7)
- Distinguished Encoding Rules (DER) Encoded Binary X.509
- Base64 Encoded X.509

3 Install this certificate on the domain controller.

The following links contain instructions for each supported platform:

- HOW TO: Install Imported Certificates on a Web Server in Windows Server 2003 (http://support.microsoft.com/default.aspx?scid=kb;en-us;816794)
- HOW TO: Install Imported Certificates on a Web Server in Windows 2000 (http://support.microsoft.com/default.aspx?scid=kb;EN-US;310178)

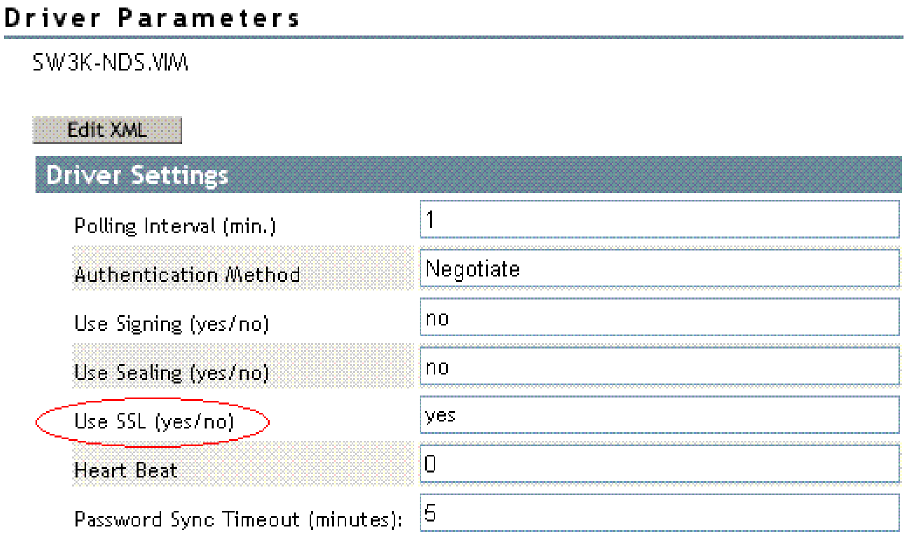
Follow the instructions listed under Import the Certificate into the Local Computer Store.

4 Ensure that a trust relationship is established between the server hosting the driver shim and the root certificate authority that issued the certificate.

The server hosting the driver shim must trust the root certificate authority that the issuing certificate authority chains to.

For more information on establishing a trust for certificates, see the *Policies to establish trust of root certification authorities* topic in *Windows 2000 Server Help*.

5 In iManager, edit the driver properties and change the *Use SSL (yes/no)* option to yes.



Driver Parameters

SW3K-NDS.VIM

Driver Settings

Polling Interval (min.)	1
Authentication Method	Negotiate
Use Signing (yes/no)	no
Use Sealing (yes/no)	no
Use SSL (yes/no)	yes
Heart Beat	0
Password Sync Timeout (minutes):	5

6 Restart the driver.

When the driver restarts, an SSL connection is negotiated between the domain controller and the server running the Active Directory driver shim.

Verify the Certificate

To verify the certificate, authenticate to Active Directory via SSL. Use the ldifde command line utility found on Windows servers. To use the ldifde command:

1 Open a command line prompt

2 Enter `ldifde -f output/input file -t 636 -b administrator domain password -s computerFullName`

Here is an example of what you would enter if your server is configured for port 636.

```
ldifde -f out.txt -t 636 -b administrator dxad.novell.com novell -s
parent1.dxad3.lab.novell
```

The output is sent to the `out.txt` file. If you open the file and see the objects in Active Directory listed, you made a successful SSL connection to Active Directory and the certificate is valid.

2.3.3 SSL Connection Between the Remote Loader and Identity Manager

If you are using the Remote Loader, you need to set up SSL between the Metadirectory engine and the Remote Loader, and configure the settings between the driver and Active Directory.

For information on establishing an SSL connection between the Remote Loader and Identity Manager, see "Providing for Secure Data Transfers" in the *Novell Identity Manager 3.5.1 Administration Guide*.

WARNING: When the Remote Loader is running on a Windows 2003 R2 SP1 32-bit server, the certificate must be in Base64 format. If you use the DER format, the Remote Loader fails to connect to the Identity Manager engine.

2.4 Creating an Administrative Account

In a test environment, use the Administrator account until you get the Active Directory driver working. Then create an administrative account that has the proper rights (including restricted rights) for the Active Directory driver to use exclusively to authenticate to Active Directory.

Doing this keeps the Identity Manager administrative account insulated from changes to other administrative accounts. Advantages to this design are:

- You can use Active Directory auditing to track the activity of the Active Directory driver.
- You can implement a password change policy as with other accounts, then make necessary updates to the driver configuration.

This account name and password are stored in the driver configuration. Therefore, you must change this password whenever the account password changes. If you change the account password without updating the driver configuration, authentication fails the next time the driver is restarted.

At a minimum, this account must have Read and Replicating Directory Changes rights at the root of the domain for the Publisher channel to operate. You also need Write rights to any object modified by the Subscriber channel. Write rights can be restricted to the containers and attributes that are written by the Subscriber channel.

To provision Exchange mailboxes, your Identity Manager account must have "Act as part of the Operating System" permission for the logon account.

Windows 2003 requires that you have additional rights in order to see deleted objects. See Appendix A, "Changing Permissions on the CN=Deleted Objects Container," on page 135.

2.5 Becoming Familiar with Driver Features

This section discusses driver features you should become familiar with before deploying the Active Directory driver.

- Section 2.5.1, "Multivalue Attributes," on page 29
- Section 2.5.2, "Managing Account Settings Using Custom Boolean Attributes," on page 29
- Section 2.5.3, "Provisioning Exchange Mailboxes," on page 30
- Section 2.5.4, "Expiring Accounts in Active Directory," on page 30
- Section 2.5.5, "Retaining eDirectory Objects When You Restore Active Directory Objects," on page 30

2.5.1 Multivalue Attributes

The way the Active Directory driver handles multivalue attributes has changed from version 2.

Version 2 treated multivalue attributes as single-valued on the Subscriber channel by ignoring all but the first change value in an Add or Modify operation. Version 3 of the Active Directory Driver fully supports multivalue attributes.

However, when the Active Directory driver synchronizes a a multivalue attribute with a single-value attribute, the multivalue attribute is treated as single-valued. For example, the Telephone Number attribute is single-valued in Active Directory, and multivalue in the Identity Vault. When this attribute is synchronized from Active Directory, only a single value is stored in the Identity Vault.

This creates true synchronization and mapping between the two attributes, but can result in a potential loss of data if you have multiple values in an attribute that is mapped to an attribute with a single value. In most cases, a policy can be implemented to preserve the extra values in another location if required in your environment.

2.5.2 Managing Account Settings Using Custom Boolean Attributes

The Active Directory attribute userAccountControl is an integer whose bits control logon account properties, such as whether logon is allowed, passwords are required, or the account is locked. Synchronizing the Boolean properties individually is problematic because each property is embedded in the integer value.

In version 2, the Active Directory driver took a shortcut that let you map userAccountControl to the eDirectory Login Disabled attribute, but didn’t let you map the other property bits within the attribute.

In version 3, each bit within the userAccountControl attribute can be referenced individually as a Boolean value, or userAccountControl can be managed in-total as an integer. The driver recognizes a Boolean alias to each bit within userAccountControl. These alias values are included in the schema for any class that includes userAccountControl. The alias values are accepted on the Subscriber channel and are presented on the Publisher channel.

The advantage to this feature is that because each bit can be used as a Boolean, the bit can be enabled individually in the Publisher filter and accessed easily. You can also put userAccountControl into the Publisher filter to receive change notification as an integer.

The integer and alias versions of userAccountControl should not be mixed in a single configuration.

The following table lists available aliases and hexadecimal values. Read-only attributes cannot be set on the Subscriber channel.

Table 2-3 *Aliases and Hexadecimal Values*

Alias	Hexadecimal	Notes
dirxml-uACDontExpirePassword	0x10000	Read-write
dirxml-uACHomedirRequired	0x0008	Read-write
dirxml-uACInterdomainTrustAccount	0x0800	Read-only

Alias	Hexadecimal	Notes
dirxml-uACNormalAccount	0x0200	Read-only
dirxml-uACServerTrustAccount	0x2000	Read-only
dirxml-uACWorkstationTrustAccount	0x1000	Read-only
dirxml-uACAccountDisable	0x0002	Read-write
dirxml-uACPasswordNotRequired	0x0020	Read-write

For troubleshooting tips relating to the userAccountControl attribute, see Section 10.8, "The Active Directory Account Is Disabled after a User Add on the Subscriber Channel," on page 118.

2.5.3 Provisioning Exchange Mailboxes

The Active Directory driver can be configured to provision Exchange accounts as well as Active Directory accounts. The Active Directory driver can provision Exchange 2000, Exchange 2003, and Exchange 2007 accounts. For information on configuring the driver to provision the Exchange mailboxes, see Appendix C, "Provisioning Exchange Accounts," on page 147.

2.5.4 Expiring Accounts in Active Directory

If you map the eDirectory attribute of Login Expiration Time to the Active Directory attribute of accountExpires, the account in Active Directory expire a day earlier than the time set in eDirectory.

This happens because Active Directory sets the value of the accountExpires attribute in full-day increments. The eDirectory attribute of Login Expiration Time uses a specific day and time to expire the account.

For example, if you set an account in eDirectory, to expire on July 15th, 2007, at 5:00 p.m., the last full day this account is valid in Active Directory is July 14th.

If you set the account in the Microsoft Management Console, to expire on July 15th, 2007, the eDirectory attribute of Login Expiration Time is set to expire on July 16th, 2007 at 12:00 a.m. Because the Microsoft Management Console doesn't allow for a value of time to be set, the default is 12:00 a.m.

The driver uses the most restrictive settings. You can add an additional day to the expiration time in Microsoft depending upon what your requirements are.

2.5.5 Retaining eDirectory Objects When You Restore Active Directory Objects

Any Active Directory objects that are restored through the Active Directory tools delete the associated eDirectory™ object when the objects are synchronized. The Active Directory driver looks for a change in the isDeleted attribute on the Active Directory object. When the driver detects a change in this attribute, a Delete event is issued through the driver for the object associated with the Active Directory object.

If you don’t want eDirectory objects deleted, you must add an additional policy to the Active Directory driver. Identity Manager 3.5 comes with a predefined rule that changes all Delete events into Remove Association events. For more information, see “Command Transformation - Publisher Delete to Disable ” in *Policies in Designer 2.1*.

Installing the Active Directory Driver

3

If you do not have an existing Active Directory driver, proceed with this section. If you have an existing driver, skip this section and proceed to Chapter 4, “Upgrading the Active Directory Driver,” on page 37.

Installing the driver adds the driver configuration file, extends the schema, and places a utility to help with the configuration of the driver. It does not create any objects. After the driver is installed, the driver object needs to be created and configured. See Chapter 5, “Configuring the Active Directory Driver,” on page 49 for instructions on how to create and configure the Active Directory driver.

The Active Directory driver can be installed locally or remotely. You need to decide which way you want to install the driver before proceeding.

- Section 3.1, “Installing the Driver Locally,” on page 33
- Section 3.2, “Installing the Driver Remotely,” on page 33
- Section 3.3, “Installing the Active Directory Discovery Tool,” on page 34

3.1 Installing the Driver Locally

If you are going to install the driver locally, the server that runs the driver is required to have the following installed:

- eDirectory™ 8.73 SP8 or above, or 8.8 SP1 or above
- Active Directory

The Active Directory driver is installed during the installation of Identity Manager. See “Installing Identity Manager” in the*Identity Manager 3.5.1 Installation Guide* for the installation instructions.

3.2 Installing the Driver Remotely

If you do not want to install eDirectory and Identity Manager on the server that has Active Directory installed on it, use the Remote Loader service. The Remote Loader service is installed and then connects to the server running Identity Manager. For installation instructions see, “Deciding Whether to Use the Remote Loader” in the *Novell Identity Manager 3.5.1 Administration Guide*.

If you decide to use the Remote Loader and install it on a member server, you must configure the driver using an SSL connection between the Remote Loader and the Identity Manager server. For more information on how to set up an SSL connection, see “Providing for Secure Data Transfers” in the *Novell Identity Manager 3.5.1 Administration Guide*.

3.3 Installing the Active Directory Discovery Tool

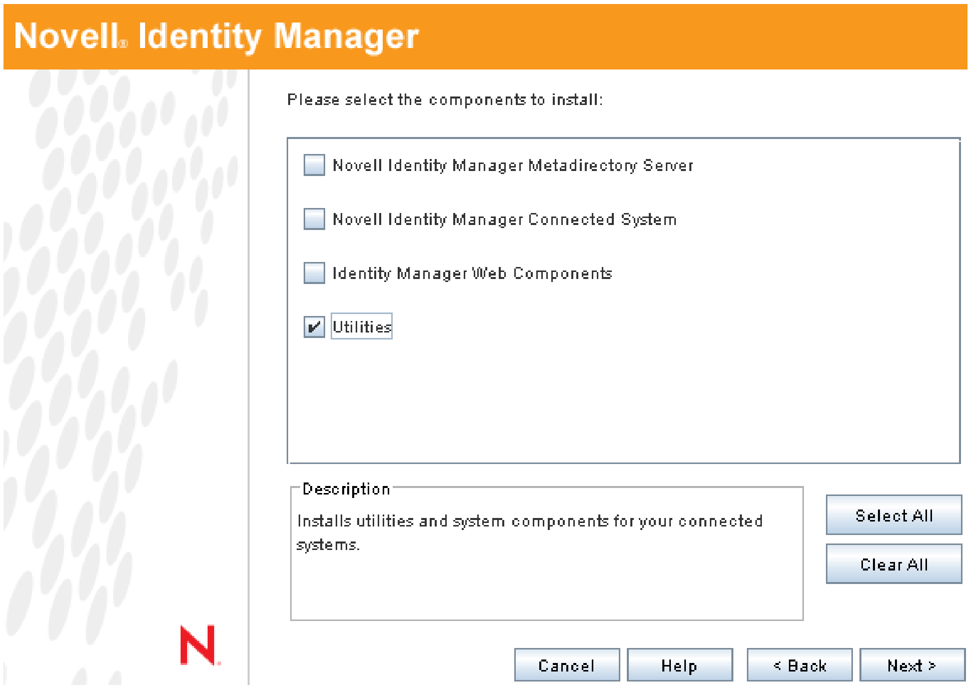
A utility that comes with Identity Manager helps gather information required to configure the driver. The utility is called the Active Directory Discovery tool. To install the Active Directory Discovery tool:

1 On the workstation that you use to configure Active Directory, launch the Identity Manager installation.

2 In the Welcome dialog box, click *Next*, then accept the license agreement.

3 In the two Identity Manager Overview dialog boxes, review the information, then click *Next*.

4 In the Please Select the Components to Install dialog box, deselect all options except *Utilities*, then click *Next*.



Novell® Identity Manager

Please select the components to install:

□ Novell Identity Manager Metadirectory Server

□ Novell Identity Manager Connected System

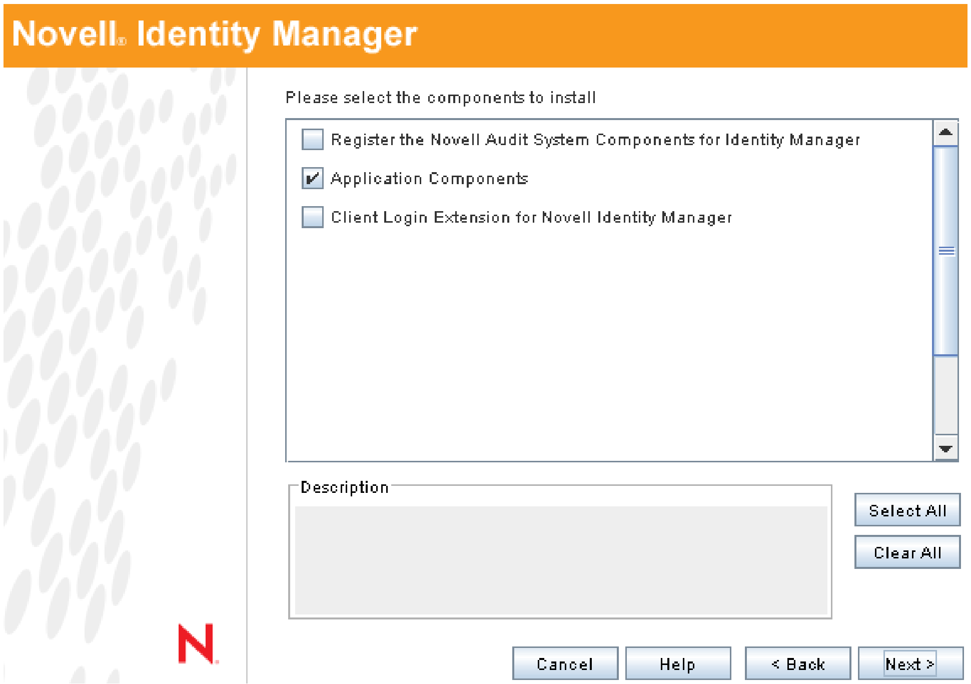
□ Identity Manager Web Components

☑ Utilities

Description

Installs utilities and system components for your connected systems.

5 Select *Application Components*, then click *Next*.



Please select the components to install

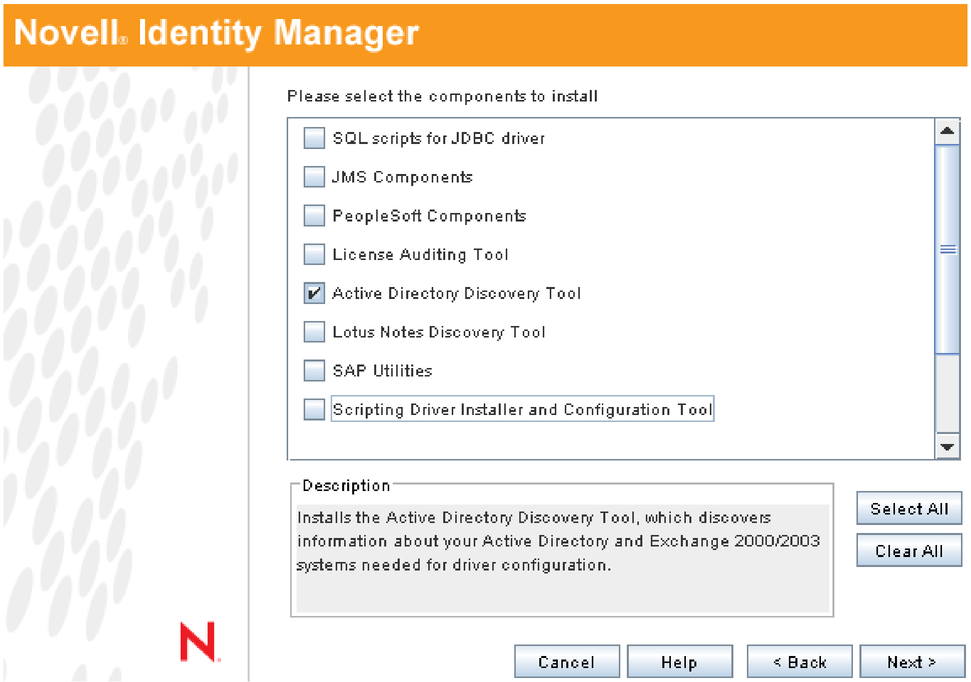
- □ Register the Novell Audit System Components for Identity Manager
- ☑ Application Components
- □ Client Login Extension for Novell Identity Manager

Description

Deselect Novell Audit System Components for Identity Manager.

6 Specify the installation path, then click *Next*.

7 Select only *Active Directory Discovery Tool*, then click *Next*.



Please select the components to install

- □ SQL scripts for JDBC driver
- □ JMS Components
- □ PeopleSoft Components
- □ License Auditing Tool
- ☑ Active Directory Discovery Tool
- □ Lotus Notes Discovery Tool
- □ SAP Utilities
- □ Scripting Driver Installer and Configuration Tool

Description

Installs the Active Directory Discovery Tool, which discovers information about your Active Directory and Exchange 2000/2003 systems needed for driver configuration.

8 Review the selected options, then click *Finish*.

9 Proceed to Chapter 5, “Configuring the Active Directory Driver,” on page 49 to create and configure the driver object.

Upgrading the Active Directory Driver

4

If you have been using a previous version of the Active Directory driver, proceed with this section. If you do not have an existing driver, do not use this section. Use Chapter 3, "Installing the Active Directory Driver," on page 33 instead.

Identity Manager 3.5 contains a new architecture for how policies reference one another. To take advantage of this new architecture, the driver configuration file provided for Active Directory must be upgraded. For more information on the new architecture, see "Upgrading Identity Manager Policies" in the *Understanding Policies for Identity Manager 3.5.1*. You can upgrade the driver in Designer or iManager.

- Section 4.1, "Checklist for Upgrading," on page 37
- Section 4.2, "Addressing the Login Disabled Value," on page 39
- Section 4.3, "Upgrading the Driver Shim from DirXML 1.1a," on page 39
- Section 4.4, "Upgrading the Driver Shim from IDM 2.x," on page 40
- Section 4.5, "Applying the Overlay for Exchange Mailboxes," on page 40
- Section 4.6, "Upgrading the Driver in Designer," on page 45
- Section 4.7, "Upgrading the Driver in iManager," on page 48

4.1 Checklist for Upgrading

To upgrade the Active Directory driver, use the following checklist. If you are not an expert with Identity Manager, you might want to engage a capable consultant.

❑ To use Identity Manager Password Synchronization, add the driver manifest and password policies.

See Upgrading an Existing Driver Configuration to Support Identity Manager Password Synchronization (http://www.novell.com/documentation/dirxml20/index.html?page=/documentation/dirxml20/admin/data/bo16oyy.html).

❑ For continued use of Password Synchronization 1.0, add legacy policies to the existing driver configuration.

See Upgrading Password Synchronization 1.0 to Password Synchronization Provided with Identity Manager (http://www.novell.com/documentation/dirxmldrivers/index.html?page=/documentation/dirxmldrivers/ad/data/bnwjt02.html).

❑ Remove the structured formatting of the sAMAccountName in the existing driver's style sheets.

sAMAccountName was a structured attribute in the DirXML® 1.1a Active Directory 2.0 driver. In the Active Directory 3.5 driver, it is a string.

Old format:

```
   <value type="structured">
    <component name="nameSpace">0</component>
    <component association-ref="XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX"
name="volume"/>
    <component name="path">jsmith</component>
   </value>
```

New format:

```
  <add-attr attr-name="sAMAccountName">
   <value type="string">jsmith</value>
  </add-attr>
```

□ Upgrade driver configuration parameters.

We recommend the use of the following settings by default:

```
<?xml version="1.0"?>
<driver-config name="Active Directory Driver">
    <driver-options>
       <pollingInterval display-name="Polling Interval (min.)">
1</pollingInterval>
       <auth-method display-name="Authentication Method">
Negotiate</auth-method>
             <signing display-name="Use Signing (yes/no)" id=">
no</signing>
             <sealing display-name="Use Sealing (yes/no)">
no</sealing>
       <use-ssl display-name="Use SSL (yes/no)">
no</use-ssl>
       <pub-heartbeat-interval display-name="Heart Beat">
0</pub-heartbeat-interval>
       <pub-password-expire-time display-name="Password Sync Timeout
       (minutes):">60</pub-password-expire-time>
       <use-CDOEXM display-name="Use CDOEXM for Exchange (yes/no)">
no</use-CDOEXM>
  <cdoexm-move display-name="Allow CDOEXM Exchange mailbox move (yes/
no)">yes</cdoexm-move>
  <cdoexm-delete display-name="Allow CDOEXM Exchange mailbox delete (yes/
no)">yes</cdoexm-delete>
    </driver-options>
</driver-config>
```

□ Convert the authentication ID to either the sAMAccountName (for example, jsmith) or the domain name/account name format (for example, *domain*/jsmith).

□ Change the mapping of the Login Disabled attribute from userAccountControl to dirxml-uACAccountDisable.

□ If you are provisioning Exchange accounts, change the driver parameter for CDOEXM to Yes, then remove the following four hard-coded attributes from your existing driver configuration style sheets:

- msExchHomeServerName
- legacyExchangeDN

- homeMTA
- msExchMailboxSecurityDescriptor

□ If you are upgrading from Identity Manager 2.*x* and you have Exchange provisioning enabled, an overlay has to be applied to the driver. Identity Manager 3.5 controls moves and deletes with the Exchange mailboxes. For this to function on an upgraded driver, the overlay must be applied. See Section 4.5, "Applying the Overlay for Exchange Mailboxes," on page 40 for information on how to apply the overlay.

4.2 Addressing the Login Disabled Value

eDirectory™ treats a lack of Login Disabled = true as being the same as Login Disabled = false. Therefore, if you install the version 3 Active Directory driver as a new installation (not an upgrade), and if a Login Disabled = false value isn't present, a default policy on the Creation Rule synthesizes that value.

Upgrading from the version 2 driver to the version 3 driver doesn't get this policy by default.

4.3 Upgrading the Driver Shim from DirXML 1.1a

The upgrade replaces the previous driver shim with the new driver shim but keeps the previous driver's configuration. The new driver shim can run the DirXML 1.1a configuration with no changes (unless you are using Password Synchronization 1.0).

If you continue to use Password Synchronization 1.0, you don't need to upgrade the driver shim. The DirXML 1.1a driver shim runs on the Identity Manager 3.5 engine, but the Identity Manager 3.5 driver shim cannot run on a DirXML 1.1a engine.

If you choose to not upgrade the driver shim, make sure that during the installation of the Identity Manager 3.5 engine, you deselect the Active Directory driver. If it is selected, the driver shim is upgraded.

To upgrade the driver shim:

1. Make sure you have updated your driver with all the patches for the version you are currently running.

   We recommend this step for all drivers, to help minimize upgrade issues.

2. Install the Identity Manager 3.5 driver shim. You can do this at the same time that you install the Identity Manager 3.5 engine.

   Follow the instructions in the "Installing Identity Manager" section in the*Identity Manager 3.5.1 Installation Guide*.

   WARNING: If you have been using Password Synchronization 1.0, don't install the upgraded Identity Manager Driver for AD until you have read Section 7.2, "Upgrading Password Synchronization 1.0 to Password Synchronization Provided with Identity Manager," on page 67 and are ready to add policies to your driver configuration to provide backward compatibility with Password Synchronization 1.0.

   Running an Identity Manager 2.0 or 3.0 driver shim or configuration with the DirXML 1.1a engine is not supported.

3 After the shim is installed, Novell eDirectory and the driver need to be restarted.

3a In iManager, click *Identity Manager* > *Identity Manager Overview*.

3b Browse to the Driver Set where the driver exists, then click *Search*.

3c Click the upper right corner of the driver icon, then click *Restart* driver.

4 Activate the driver shim with your Identity Manager activation credentials.

See Chapter 6, "Activating the Driver," on page 63.

After you install the driver shim, continue with Chapter 5, "Configuring the Active Directory Driver," on page 49.

4.4 Upgrading the Driver Shim from IDM 2.*x*

1 Make sure you have updated your driver with all the patches for the version you are currently running.

We recommend this step for all drivers, to help minimize upgrade issues.

2 Install the Identity Manager 3.5 driver shim. You can do this at the same time that you install the Identity Manager 3.5 engine.

Follow the instructions in "Installing Identity Manager" section found in the *Identity Manager 3.5.1 Installation Guide*.

WARNING: If you have been using Password Synchronization 1.0, don't install the upgraded Identity Manager Driver for AD until you have read Section 7.2, "Upgrading Password Synchronization 1.0 to Password Synchronization Provided with Identity Manager," on page 67 and are ready to add policies to your driver configuration to provide backward compatibility with Password Synchronization 1.0.

Running an Identity Manager driver shim or configuration with the DirXML 1.1a engine is not supported.

3 After the shim is installed, Novell eDirectory and the driver need to be restarted. Follow the instructions in Section 8.3, "Starting, Stopping, or Restarting the Driver," on page 91.

4 Activate the driver shim with your Identity Manager activation credentials.

See Chapter 6, "Activating the Driver," on page 63.

After you install the driver shim, continue with Chapter 5, "Configuring the Active Directory Driver," on page 49.

4.5 Applying the Overlay for Exchange Mailboxes

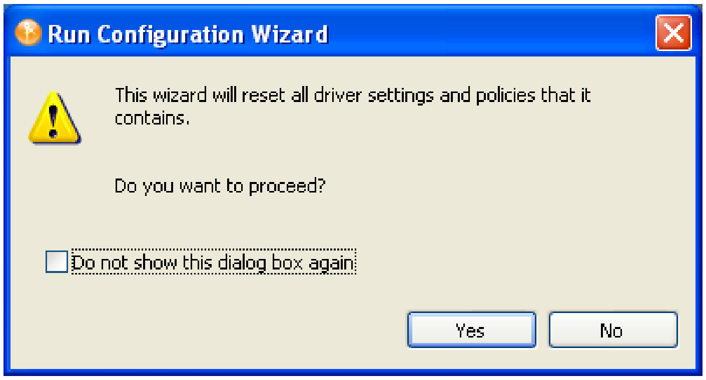
If you have upgraded from Identity Manager 2.*x* to Identity Manager 3.0.1 or above, the Active Directory driver overlay needs to be applied if Exchange provisioning is enabled on the driver. The overlay allows the driver to control deletes and moves with the Exchange mailboxes.

- Section 4.5.1, "Applying the Overlay in Designer," on page 41
- Section 4.5.2, "Applying the Overlay in iManager," on page 43

4.5.1 Applying the Overlay in Designer

1 In the modeler, right-click on the AD driver connector icon, then click *Run Configuration Wizard*.

2 Click *Yes* to allow the Configuration Wizard to run.



Run Configuration Wizard

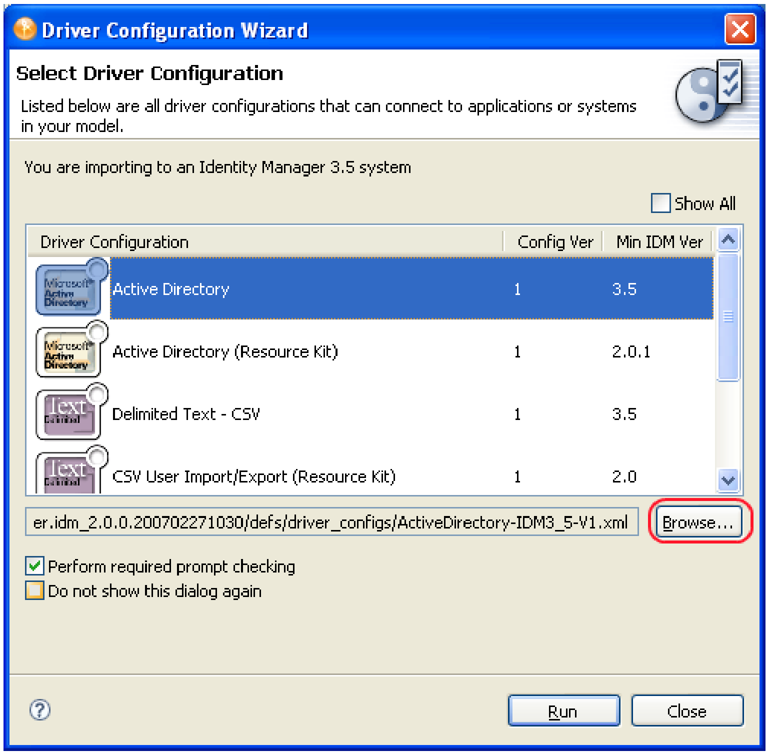
This wizard will reset all driver settings and policies that it contains.

Do you want to proceed?

□ Do not show this dialog box again

3 Select *Browse* and browse to the ActiveDirectoryUpdate.xml file, then click *Open*.

The file is located in the following plug-in: eclipse\plugins\com.novell.designer.idm_*x.x.x*\defs\driver_configs\overlay_configs\ActiveDirectoryUpdate.xml.



Driver Configuration Wizard

Select Driver Configuration

Listed below are all driver configurations that can connect to applications or systems in your model.

You are importing to an Identity Manager 3.5 system

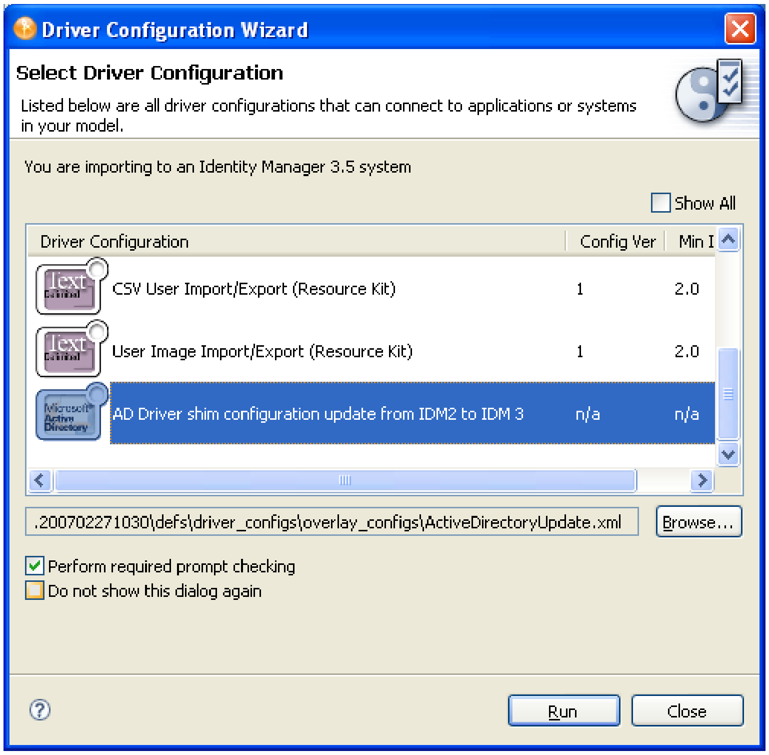
□ Show All

Driver Configuration	Config Ver	Min IDM Ver
Active Directory	1	3.5
Active Directory (Resource Kit)	1	2.0.1
Delimited Text - CSV	1	3.5
CSV User Import/Export (Resource Kit)	1	2.0

er.idm_2.0.0.200702271030/defs/driver_configs/ActiveDirectory-IDM3_5-V1.xml

☑ Perform required prompt checking
□ Do not show this dialog again

4 Select *AD Driver shim configuration update from IDM2 to IDM3*, then click *Run*.


Select Driver Configuration

Listed below are all driver configurations that can connect to applications or systems in your model.

You are importing to an Identity Manager 3.5 system

☐ Show All

Driver Configuration	Config Ver	Min I
CSV User Import/Export (Resource Kit)	1	2.0
User Image Import/Export (Resource Kit)	1	2.0
AD Driver shim configuration update from IDM2 to IDM 3	n/a	n/a

.200702271030\defs\driver_configs\overlay_configs\ActiveDirectoryUpdate.xml

☑ Perform required prompt checking

☐ Do not show this dialog again

5 Provide the information specific to your environment, then click *Next*. See Table 4-1 on page 42 for a description of the fields.

6 Click *OK* on the result screen. Review this information for any errors.

Table 4-1 *Overlay Configuration Parameters in Designer*

Parameter	Description
Driver name	The driver that needs to be updated with the new parameters. Enter in the driver name or browse to and select the driver.
Update driver	Updates the driver with the parameters. Select *Yes* if you want the driver updated. Select *No* if you do not want to update the driver.
homeMDB controls Exchange move	Allows a change to the user HomeMDB attribute to result in a move on the user's Exchange mailbox when using CDOEXM. The Exchange Message Database, where the user's mailbox is moved to, must be in the same domain as the old Exchange Message Database. If *Yes* is selected and a User object is moved in eDirectory, the move is reflected in Active Directory and Exchange as well. If *No* is selected and a User object is moved in eDirectory, it is reflected in Active Directory, but not in Exchange.

Parameter	Description
homeMDB controls Exchange delete	Allows removal of the user homeMDB attribute to result in a delete of the user's Exchange mailbox when using CDOEXM. If *Yes* is selected and an eDirectory User object is deleted, the associated Active Directory User object and Exchange accounts are deleted. If *No* is selected and an eDirectory User object is deleted, the associated Active Directory User object is deleted, but the Exchange account is left intact.
Logon and impersonate	Allows the driver authentication account for CDOEXM and Password Set support to logon in different manners. If *No* is selected, the driver performs only a network logon. If *Yes* is selected, the driver performs a local logon. The authentication account must be an Active Directory account with administrative privileges.

4.5.2 Applying the Overlay in iManager

There are two different ways to update the driver through iManager. It can be updated in the Identity Manager Overview or through Identity Manager Utilities.

- "Identity Manager Overview" on page 43
- "Identity Manager Utilities" on page 44

Identity Manager Overview

1 In iManager, select *Identity Manager* > *Identity Manager Overview*.

2 Select *Search* to find the Driver Set object where the Active Directory driver is stored.

3 Select *Add Driver* in the Identity Manager Overview page.

4 Browse to and select the Driver Set object where the Active Directory driver is stored, then click *Next*.

5 Select *Import a driver configuration from the server (.XML file)*.

6 From the drop-down menu, select *ActiveDirectoryUpdate.xml*, then click *Next*.

7 Provide the information specific to your environment, then click *Next*. See Table 4-2 on page 43 for a description of the fields.

8 Select *Update that driver and all policy libraries* to update the driver, or select *Specify a different name for the driver and/or location for the policy libraries*, then click *Next*.

9 View the summary of changes, then click *Finish*.

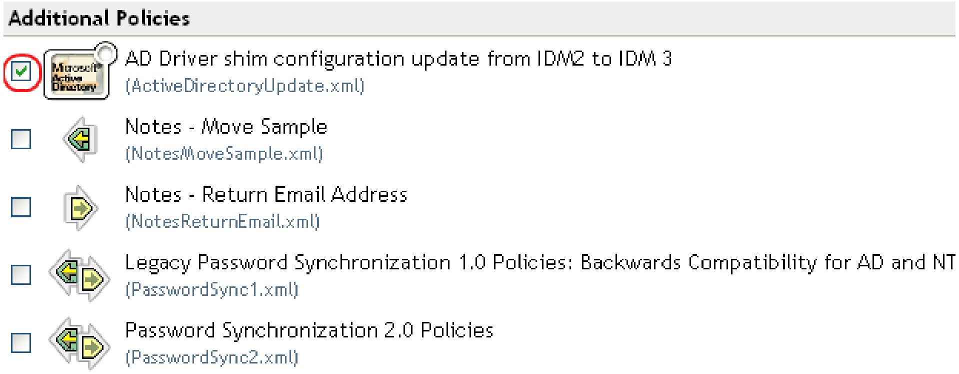
Table 4-2 *Overlay Configuration Parameters in iManager*

Parameter	Description
Driver name	The driver that needs to be updated with the new parameters.

Parameter	Description
Existing drivers	From the drop-down menu, select the name of the updated AD driver with Exchange provisioning enabled. After the driver name is selected, the Drive name field is automatically populated.
Update driver	Updates the driver with the parameters. Select *Yes* if you want the driver updated. Select *No* if you do not want to update the driver.
homeMDB controls Exchange move	Allows a change to the user HomeMDB attribute to result in a move on the user's Exchange mailbox when using CDOEXM. The Exchange Message Database, where the user's mailbox is moved to, must be in the same domain as the old Exchange Message Database. If *Yes* is selected and a User object is moved in eDirectory, the move is reflected in Active Directory and Exchange as well. If *No* is selected and a User object is moved in eDirectory, it is reflected in Active Directory, but not in Exchange.
homeMDB controls Exchange delete	Allows removal of the user homeMDB attribute to result in a delete of the user's Exchange mailbox when using CDOEXM. If *Yes* is selected and an eDirectory User object is deleted, the associated Active Directory User object and Exchange accounts are deleted. If *No* is selected and an eDirectory User object is deleted, the associated Active Directory User object is deleted, but the Exchange account is left intact.
Logon and impersonate	Allows the driver authentication account for CDOEXM and Password Set support to logon in different manners. If *No* is selected, the driver performs only a network logon. If *Yes* is selected, the driver performs a local logon. The authentication account must be an Active Directory account with administrative privileges.

Identity Manager Utilities

1. In iManager, select *Identity Manager Utilities > Import Configurations*.
2. Browse to and select the Driver Set object where the Active Directory driver is stored, then click *Next*.
3. Under *Additional Policies*, select *AD Driver shim configuration update from IDM2 to IDM 3*, then click *Next*.



Additional Policies

☑ AD Driver shim configuration update from IDM2 to IDM 3 (ActiveDirectoryUpdate.xml)

□ Notes - Move Sample (NotesMoveSample.xml)

□ Notes - Return Email Address (NotesReturnEmail.xml)

□ Legacy Password Synchronization 1.0 Policies: Backwards Compatibility for AD and NT (PasswordSync1.xml)

□ Password Synchronization 2.0 Policies (PasswordSync2.xml)

4 Provide the information specific to your environment, then click *Next*. See Table 4-2 on page 43 for a description of the fields.

5 Select *Update that driver and all policy libraries* to update the driver, or select *Specify a different name for the driver and/or location for the policy libraries*, then click *Next*.

6 View the summary of changes, then click *Finish*.

4.6 Upgrading the Driver in Designer

1 Make sure you have updated your driver with all the patches for the version you are currently running.

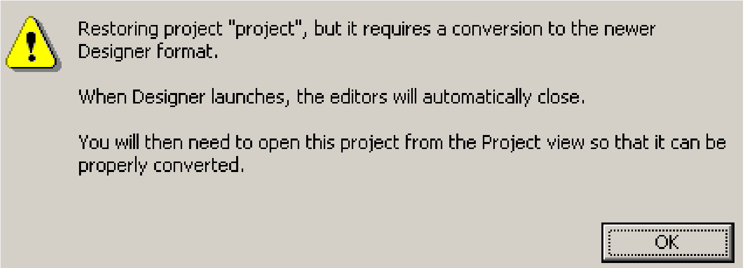
We recommend this step for all drivers, to help minimize upgrade issues.

2 Back up the driver. See Chapter 11, “Backing Up the Driver,” on page 129 for instruction on how to back up the driver.

3 Install Designer version 2.0 or above, then launch Designer.

If you had a project open in Designer when you upgraded Designer, proceed to Step 4. If you didn’t have a project open in Designer when you upgraded Designer, skip to Step 5.

4 If you had a project open when upgrading Designer, the following warning message is displayed. Read the warning message, then click *OK*.



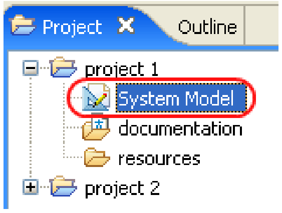
Restoring project "project", but it requires a conversion to the newer Designer format.

When Designer launches, the editors will automatically close.

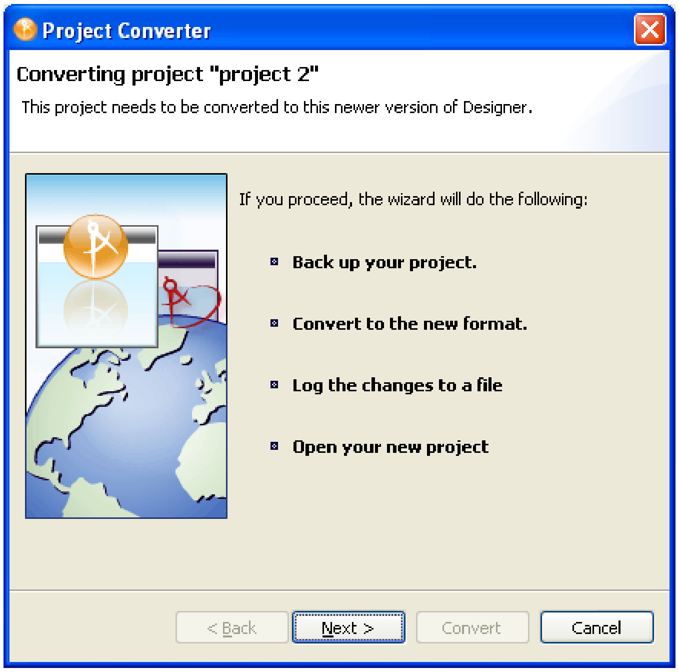
You will then need to open this project from the Project view so that it can be properly converted.

Designer closes the project to preform the upgrade.

5 In the Project view, double-click *System Model* to open and convert the project.



6 Read the Project Converter message explaining the conversion process, then click *Next*.



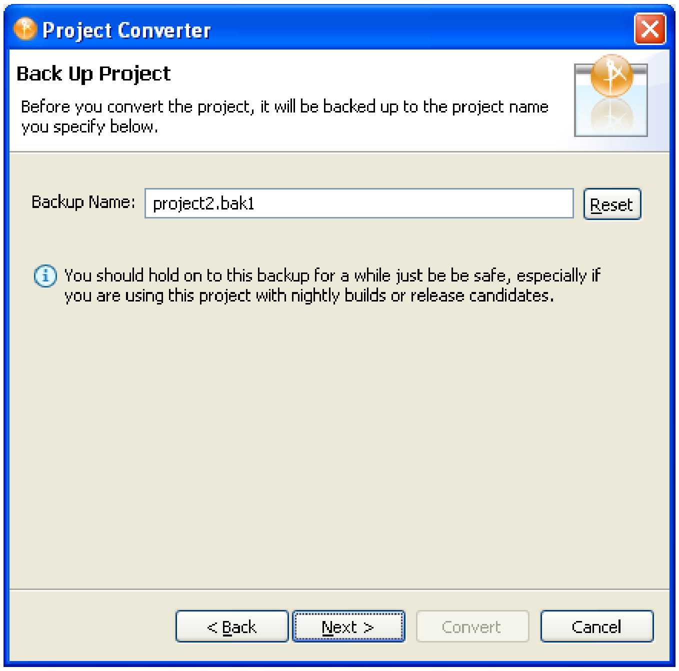
Converting project "project 2"

This project needs to be converted to this newer version of Designer.

If you proceed, the wizard will do the following:

- Back up your project.
- Convert to the new format.
- Log the changes to a file
- Open your new project

7 Specify the name of the backup project name, then click *Next*.



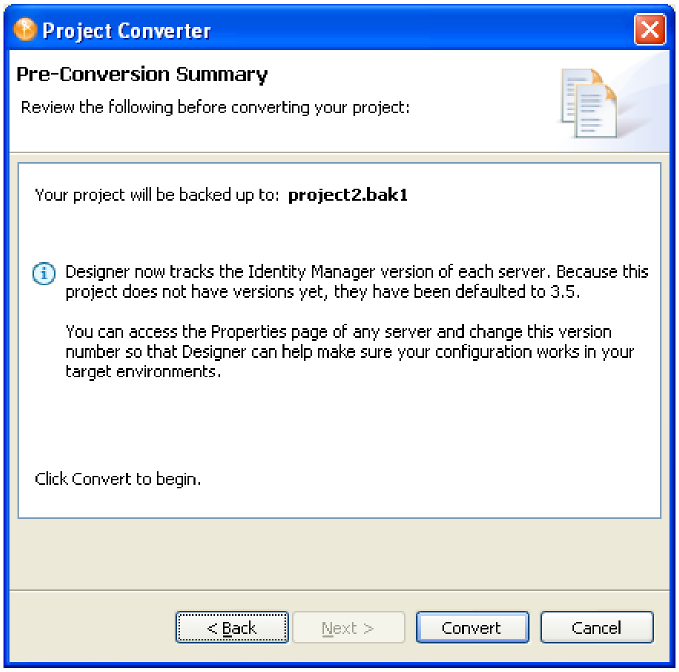
Back Up Project

Before you convert the project, it will be backed up to the project name you specify below.

Backup Name: project2.bak1

You should hold on to this backup for a while just be be safe, especially if you are using this project with nightly builds or release candidates.

8 Read the project conversion summary, then click *Convert*.



Pre-Conversion Summary

Review the following before converting your project:

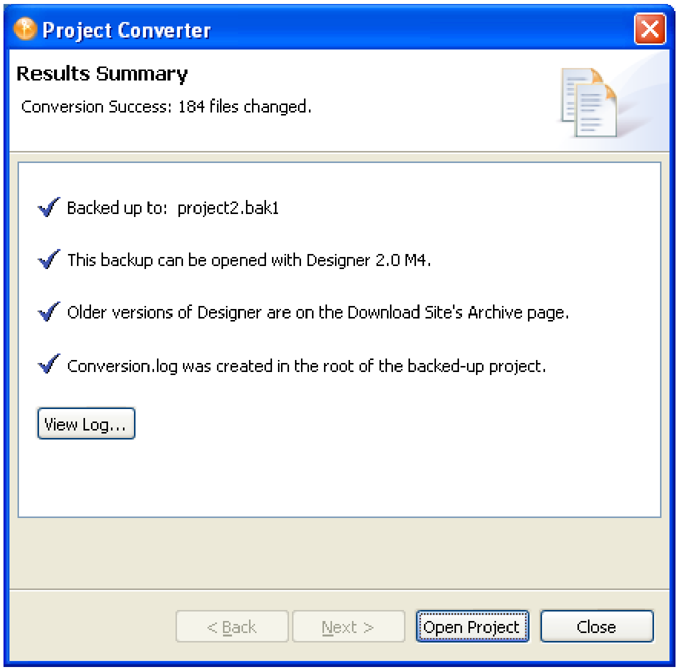
Your project will be backed up to: **project2.bak1**

Designer now tracks the Identity Manager version of each server. Because this project does not have versions yet, they have been defaulted to 3.5.

You can access the Properties page of any server and change this version number so that Designer can help make sure your configuration works in your target environments.

Click Convert to begin.

9 Read the project conversion result summary, then click *Open Project*.



Results Summary

Conversion Success: 184 files changed.

✓ Backed up to: project2.bak1

✓ This backup can be opened with Designer 2.0 M4.

✓ Older versions of Designer are on the Download Site's Archive page.

✓ Conversion.log was created in the root of the backed-up project.

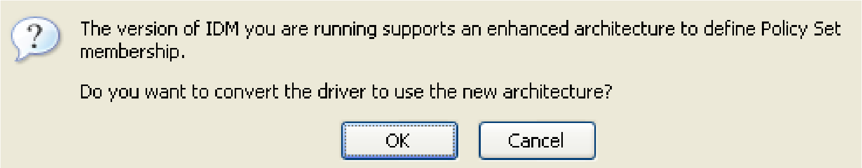
If you want to view the log file that is generated, click *View Log*.

4.7 Upgrading the Driver in iManager

1. Make sure you have updated your driver with all the patches for the version you are currently running.

   We recommend this step for all drivers, to help minimize upgrade issues.

2. Back up the driver. See Chapter 11, “Backing Up the Driver,” on page 129 for instruction on how to back up the driver.
3. Verify that Identity Manager 3.5 has be installed and you have the current plug-ins installed, then launch iManager.
4. Click *Identity Manager* > *Identity Manager Overview*.
5. Click *Search* to find the Driver Set object, then click the driver you want to upgrade.
6. Read the message that is displayed, then click *OK*.



The version of IDM you are running supports an enhanced architecture to define Policy Set membership.

Do you want to convert the driver to use the new architecture?

7. If there is more than one driver to upgrade, repeat Step 2 through Step 6.

Configuring the Active Directory Driver

5

After the Active Directory driver files are installed, a driver object needs to be created and configured. This is done through Designer or iManager. A basic driver is created through the configuration file and it can be customized to fit your business needs.

The driver is customized by using policies. When you customize the driver, you need to know that the driver is case sensitive and you should follow Active Directory schema naming conventions when referencing objects and attributes in the Active Directory namespace. For more information about policies, see "Understanding Policies for Identity Manager 3.5.1".

To know more about credential provisioning policies, see "Managing Novell Credential Provisioning Policies".

This section explains:

5.1 Using the Active Directory Discovery Tool

The Active Directory Discovery Tool gathers the information needed to configure the Active Directory driver. The tool gathers a list of the domain controllers and Microsoft Exchange private message stores available in the domain and optionally creates an account in Active Directory suitable for the driver.

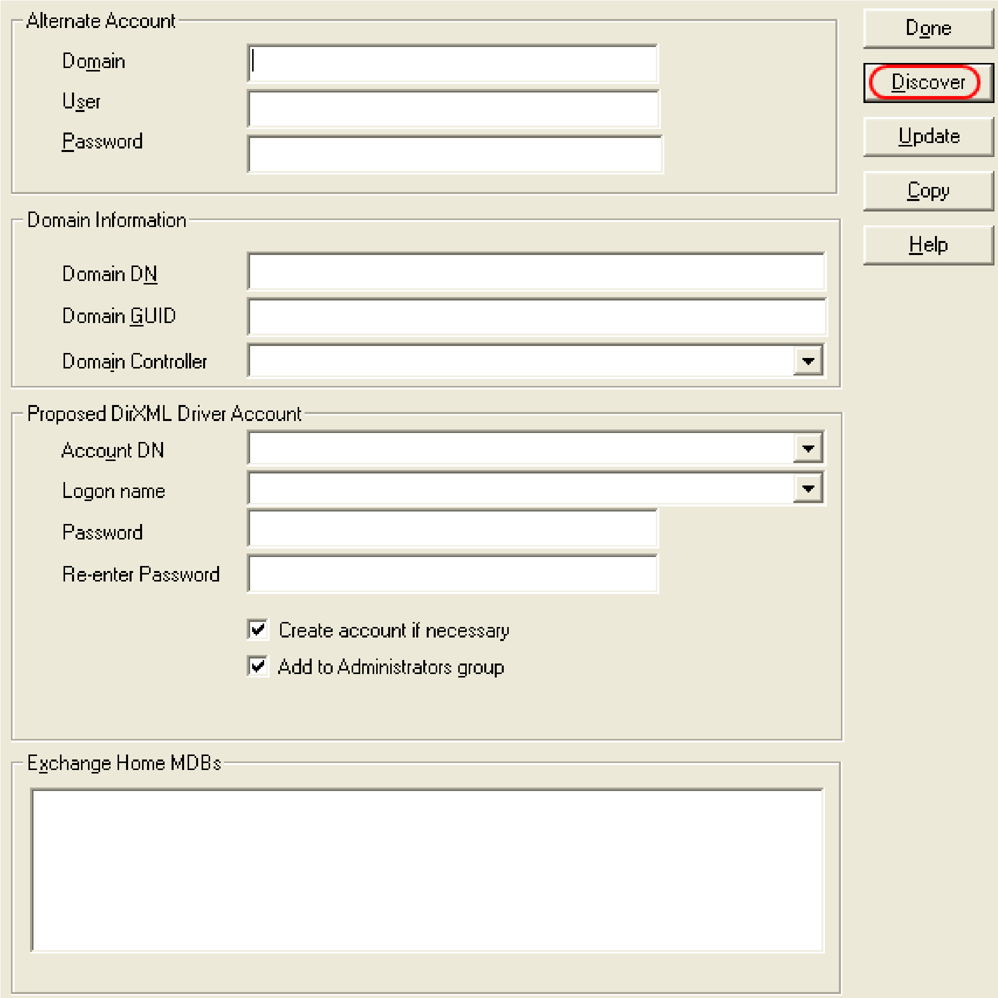
To run the tool:

1. Double-click the file `C:\Novell\NDS\DirXMLUtilities\ad_disc\ADManager.exe`.

   This is the default installation location for the file.
2. Click *Discover* to populate the tool with your domain information.

   The tool lists the following information:

   - Domain DN
   - Domain GUID
   - Domain Controller name
   - Proposed driver account name and password
   - Exchange Home MDB attribute



☑ Create account if necessary

☑ Add to Administrators group

3 If you want to see information for another domain, specify the domain name, a user with sufficient rights to look up domain information, and that user's password, then click *Discover*.

5.2 Importing the Driver Configuration File in Designer

Designer allows you to import the basic driver configuration file for Active Directory. This file creates and configures the objects and policies needed to make the driver work properly. The following instructions explain how to create the driver and import the driver's configuration.

There are many different ways of importing the driver configuration file. This procedure only documents one way.

1 Open a project in Designer. In the Modeler, right-click the driver set and select *New* > *Driver*.

2 From the drop-down list, select *Active Directory 3.5.1,* then click *Run*.

3 Configure the driver by filling in the fields. Specify information for your environment.

For information on the settings, see Section 5.4, "Default Configuration Parameters," on page 52.

4 After specifying parameters, click *Finish* to import the driver.

5 After the driver is imported, customize and test the driver.

6 After the driver is fully tested, deploy the driver into the Identity Vault.

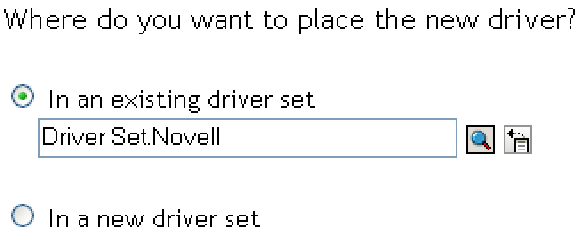
See "Deploying a Driver to an Identity Vault" in the *Designer 2.1 for Identity Manager 3.5.1*.

5.3 Importing the Driver Configuration File in iManager

The Active Directory preconfiguration file is an example configuration file. You installed this file when you installed the Identity Manager Web components on an iManager server. You use the preconfiguration file as a template that you import and customize or configure for your environment.

1 In iManager, select *Identity Manager Utilities > Import Configuration*.

2 Select a driver set, then click *Next*.



Where do you want to place the new driver?

● In an existing driver set

Driver Set.Novell

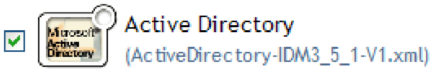
○ In a new driver set

If you place this driver in a new driver set, you must specify a driver set name, context, and associated server.

3 Select how you want the driver configurations sorted:

- All configurations
- Identity Manager 3.5 configurations
- Identity Manager 3.0 configurations
- Configurations not associated with an IDM version

4 Select the *Active Directory-IDM3_5_1-V1.xml* driver, then click *Next*.



☑ Active Directory (ActiveDirectory-IDM3_5_1-V1.xml)

5 Configure the driver by filling in the configuration parameters, then click *Next*. For information on the settings, see Section 5.4, "Default Configuration Parameters," on page 52.

6 Define security equivalences, using a user object that has the rights that the driver needs to have on the server, then click *OK*.

The tendency is to use the Admin user object for this task. However, you might want to create a DriversUser (for example) and assign security equivalence to that user. Whatever rights that the driver needs to have on the server, the DriversUser object must have the same security rights.

7 Identify all objects that represent administrative roles and exclude them from replication, then click *OK*.

Exclude the security-equivalence object (for example, DriversUser) that you specified in Step 6. If you delete the security-equivalence object, you have removed the rights from the driver, and the driver can't make changes to Identity Manager.

8 Click *Finish*.

5.4 Default Configuration Parameters

The following table explains the parameters you must provide during initial driver configuration.

NOTE: The parameters are presented on multiple screens. Some parameters are only displayed if the answer to a previous prompt requires more information to properly configure the policy.

Table 5-1 *Default Configuration Parameters*

Field	Description
Driver name	The object name to be assigned to this driver. Because each Active Directory domain requires a separate driver, you should include the domain name in the driver name. When you look at the driver, you can see which domain it is associated with.
Authentication Method	The method to authenticate with Active Directory. *Negotiate* is the preferred method. Select *Negotiate* to use the Microsoft security package to negotiate authentication. To use *Negotiate*, the server hosting the driver must be a member of the domain. If you plan to use password synchronization and are running on a member server, you need SSL. *Simple* uses an LDAP simple bind. If you select *Simple*, SSL is recommended. **IMPORTANT:** Simple bind doesn't support password synchronization or Exchange provisioning.
Authentication Id	An Active Directory account with administrative privileges to be used by Identity Manager. The name form used depends on the selected authentication mechanism. See Section 5.1, "Using the Active Directory Discovery Tool," on page 49 for more information. For *Negotiate*, provide the name form required by your Active Directory authentication mechanism. For example: ◆ Administrator - AD Logon Name ◆ Domain/Administrator - Domain qualified AD Logon Name For *Simple*, provide an LDAP ID. For example: ◆ cn=DirXML,cn=Users,DC=domain,dc=com **NOTE:** The driver must have an *Authentication Id* specified. It is a required field.

Field	Description
Authentication Password	The password for the user account specified in Authentication ID.
Authentication Context	The name of the Active Directory domain controller to use for synchronization. See Section 5.1, “Using the Active Directory Discovery Tool,” on page 49 for more information. For example, for the *Negotiate* authentication method, use the DNS name mycontroller.domain.com. For the *Simple* authentication method, you can use the IP address of your server (for example, 10.10.128.23 or the DNS name). If no value is specified, localhost is used. **NOTE:** This value is stored in the Authentication Context attribute. To change this value after the initial configuration, modify this attribute as explained in “Default Configuration of the Security Parameters” on page 131.
Domain Name	The Active Directory domain managed by this driver. See Section 5.1, “Using the Active Directory Discovery Tool,” on page 49 for more information. The driver requires LDAP formatted domain names: `dc=domain,dc=com`
Domain DNS Name	The DNS name of the Active Directory domain managed by this driver. The driver requires DNS formatted domain names: `domain.com`
Driver Polling Interval	The Identity Vault sends changes to Active Directory as they happen. However, changes to Active Directory are sent to the Identity Vault only as often as the configured polling interval. The default is 1 minute. **IMPORTANT:** The polling interval affects system performance. A low polling interval results in frequent searches and fast updates of data. A high polling interval results in periodic bursts of traffic. Although a low polling interval has a greater overall cost, the cost is spread more evenly over time. If you set the interval to 0 (zero), you get a ten-second poll rate.

Field	Description
Password Sync Timeout (minutes)	The number of minutes the driver attempts to synchronize a password, if the first attempt fails. Set the value large enough to handle whatever temporary backlog of passwords exists. If you are doing bulk changes, set the timeout large enough to handle all the changes. The rule of thumb is to allow one second per password. For example, to synchronize 18,000 passwords, allow 300 minutes (18,000 passwords divided by 60 seconds). A setting of -1 is indefinite. Although this setting can handle bulk changes, it can cause problems. For example, a password might never be able to synchronize because the account wasn't associated. Such a password would therefore remain in the system forever. A number of similar situations could result in a large inventory of unsynchronized passwords held by the system. Setting the value to 0 disables password synchronization for the driver. For more information, see Section 7.8, "Disabling Password Synchronization on a Driver," on page 87. You must set the password sync timeout to at least three times the polling interval.
Driver is Local/Remote	Configure the driver for use with the Remote Loader service by selecting *Remote*, or select *Local* to configure the driver for local use.
Remote Host Name and Port	Remote option only. The host name or IP address and port number where the Remote Loader Service has been installed and is running for this driver. The default port is 8090. This setting displays only if you set *Driver is Local/Remote* to *Remote*.
Driver Password	Remote option only. The Remote Loader uses the Driver Object Password to authenticate itself to the Identity Manager server. The password must be the same password that is specified as the Driver object password on the Remote Loader. This setting displays only if you set *Driver is Local/Remote* to *Remote*.
Remote Password	Remote option only. The Remote Loader password is used to control access to the Remote Loader instance. The password must be the same password that is specified as the Remote Loader password on the Remote Loader. This setting displays only if you set *Driver is Local/Remote* to *Remote*.

Field	Description
Base container in eDirectory	Specify the base container in the Identity Vault for synchronization. This container is used in the Subscriber Matching policies to limit the Identity Vault objects being synchronized and in the Publisher Placement policies when adding objects to the Identity Vault. New users are placed in this container by default. Use the dot format. For example, `users.myorg` If the container doesn't exist, you must create it and make sure it is associated with the Active Directory base container before trying to add users to this container.
Publisher Placement	*Mirrored* places objects hierarchically within the base container. *Flat* places objects strictly within the base container. This selection builds the default Publisher Placement policies. **NOTE:** If you select *Mirrored*, the driver assumes that the structure of the eDirectory™ database is the same in Active Directory as the eDirectory base container. If the structure is not the same, the objects are not placed properly. Create the same structure in Active Directory that exists in eDirectory, or migrate the eDirectory containers before migrating User objects.

Base container in Active Directory	Specify the base container in Active Directory, in LDAP format. New users are placed in this container by default. For example,
	`CN=Users,DC=MyDomain,DC=com`
	If the target container doesn't exist, you must create it and make sure it is associated with the eDirectory base container before trying to add users to this container.
	If you are creating or using a container other than Users in Active Directory, the container is an OU, not a CN. For example,
	`OU=Sales,OU=South,DC=MyDomain,DC=com`

Active Directory Placement	*Mirrored* places the objects hierarchically within the base container. *Flat* places objects strictly within the base container. This selection builds the default Subscriber Placement policies. **NOTE:** If you select *Mirrored*, the driver assumes that the structure of the Active Directory database is the same in eDirectory as the Active Directory base container. If the structure is not the same, the objects are not placed properly. Create the same structure in eDirectory that exists in Active Directory, or migrate the Active Directory containers before migrating User objects.

Field	Description
Configure Data Flow	Establishes the initial driver filter that controls the classes and attributes that will be synchronized. The purpose of this option is to configure the driver to best express your general data flow policy. It can be changed after import to reflect specific requirements. *Bidirectional* sets classes and attributes to synchronize on both the Publisher and Subscriber channels. A change in either the Identity Vault or Active Directory is reflected on the other side. Use this option if you want both sides to be authoritative sources of data. *AD to Vault* sets class and attributes to synchronize on the Publisher channel only. A change in Active Directory is reflected in the Identity Vault, but Identity Vault changes are ignored. Use this option if you want Active Directory to be the authoritative source of data. *Vault to AD* sets classes and attributes to synchronize on the Subscriber channel only. A change in the Identity Vault is reflected in Active Directory, but Active Directory changes are ignored. Use this option if you want the vault to be the authoritative source of data. **IMPORTANT:** Delete, Move, and Rename events are independent of the filter. It does not matter which option you select, these events are processed by the driver. If you do not want these events to synchronize, you must change the default configuration of the driver. You can use one of the predefined policies that comes with Identity Manager 3.5.1 to change Delete events into Remove Association events. For more information, see "Command Transformation - Publisher Delete to Disable " in the *Policies in Designer 2.1*. To block Move and Rename events, you must customize the driver.
Password Failure Notification User	Password synchronization policies are configured to send e-mail notifications to the associated user when password updates fail. You have the option of sending a copy of the notification e-mail to another user, such as a security administrator. If you want to send a copy, enter or browse for the DN of that user. Otherwise, leave this field blank.
Configure Entitlements	The driver can be configured to use Entitlements to manage user accounts and group memberships in Active Directory and to provision Exchange mailboxes. When using Entitlements, the driver works in conjunction with external services such as the Identity Manager User Application or Role-Based Entitlements to control the conditions under which these features are provisioned or de-provisioned in Active Directory. See "Entitlements" on page 19 for more information. Select *Yes* if you plan to use one of these external services to control provisioning to Active Directory. Select *No* if you do not plan on using the Identity Manager User Application or provisioning Exchange mailboxes.

Field	Description
User account policy	Configure Entitlements option only. User accounts in Active Directory can be controlled by synchronization or by using Entitlements with the Workflow service or Role-Based Entitlements. *Entitlements* gives control of enabling accounts in Active Directory to the Entitlement in the Identity Vault. *Implement in policy* uses the policies in the driver instead of Entitlements.
Exchange policy	Configure Entitlements option only. Exchange provisioning can be handled by driver policy, Entitlements, or skipped entirely. A user can be assigned a mailbox in Exchange (the user is mailbox enabled) or have information about a foreign mailbox stored in the Identity Vault record (the user is mail enabled). When using the driver policy, the decision to mailbox enable or mail enable a user, plus the Exchange message database where the account will reside, is controlled completely in the policy. When using *Entitlements*, an external service such as the Workflow service or Role-Based Entitlements makes these decisions and driver policy simply applies them. *Implement in policy* uses the policies in the driver instead of Entitlements to assign Exchange mailboxes. When *None* is selected, the default configuration does not create Exchange mailboxes but does synchronize the Identity Vault Internet E-Mail Address with the Active Directory mail attribute.
Group membership policy	Configure Entitlements only. Group membership in Active Directory can be controlled by synchronizing the membership list or by using Entitlements. *Entitlements* uses the Workflow service or the Role-Based Entitlements to assign group membership. *Synchronize* uses policies to synchronize the group membership list. *None* does not synchronize group membership information.
Exchange Management interface type	For more information about configuring the driver to synchronize Exchange Accounts, see Appendix C, “Provisioning Exchange Accounts,” on page 147. *use-cdoexm* synchronizes Exchange 2000 and Exchange 2003 accounts. *use-post-cdoexm* synchronizes Exchange 2007 accounts.
Allow Exchange mailbox move (yes/no)	Exchange Policy option only > Implement in policy option only. When enabled, the driver shim intercepts modifications to the Active Directory homeMDB attribute and calls into CDOEXM to move the mailbox to the new message data store. *Yes* moves the Exchange mailbox. *No* does not move the Exchange mailbox.

Field	Description
Allow Exchange mailbox delete (yes/no)	Exchange Policy option only > Implement in policy option only. When enabled, the driver shim intercepts removal for the Active Directory homeMDB attribute and calls into CDOEXM to delete the mailbox. *Yes* allows the Exchange mailbox to be deleted. *No* does not allow the Exchange mailbox to be deleted.
Default Exchange MDB	Exchange Policy > Implement in policy option only. Specifies the default Exchange Message Database (MDB). To obtain the correct name for the Exchange MDB, see Section 5.1, “Using the Active Directory Discovery Tool,” on page 49. For example, `[CN=Mailbox Store (CONTROLLER),CN=First Storage Group,CN=InformationStore,CN=CONTROLLER,CN=Servers,CN=First Administrative Group,CN=Administrative Groups,CN=Domain,CN=Microsoft Exchange,CN=Services,CN=Configuration,DC=Domain,DC=com]` The driver can be updated to manage additional MDBs after the import is complete.
When account entitlement revoked	Exchange Policy option only. Allows you to choose what action is taken when a User account is removed by Entitlements. *Disable Account* *Delete Account*
Name mapping policy selection	The driver maps the Identity Vault Full Name attribute to the Active Directory object name and maps the Active Directory Pre-windows 2000 logon name to the Identity Vault user name. You can accept the full policy or manually select parts of the policy. If the policy does not meet your needs, you can modify it after import by editing the NameMap policies in the Subscriber and Publisher Command Transformation policies after the import completes. *Accept* uses the full policy. *Manual* allows you to use part of the policy.
Full Name Mapping	Name mapping policy selection > Manual option only. *Yes* allows the driver to keep the Identity Vault Full Name attribute synchronized with the Active Directory object name and display name. *No* does not keep the Identity Vault Full Name attribute synchronized with the Active Directory object name and display name. This policy is useful when creating user accounts in Active Directory by using the Microsoft Management Console Users and Computers snap-in.

Field	Description
Logon Name Mapping	Name mapping policy selection > Manual option only. *Yes* allows the driver to keep the Identity Vault object name synchronized with the Active Directory Pre-Windows 2000 Logon Name (also known as the NT Logon Name and the sAMAccountName). *No* does not keep the Identity Vault object name synchronized with the Active Directory Pre-Windows 2000 Logon Name.
Import will proceed to Active Directory logon name policy selections	Name mapping policy selection > Manual option only. *OK*
User Principal Name Mapping	Allows you to choose a method for managing the Active Directory Windows 2000 Logon Name (also known as the userPrincipalName). userPrincipalName takes the form of an e-mail address, such as usere@domain.com. Although the shim can place any value into userPrincipalName, it is not useful as a logon name unless the domain is configured to accept the domain name used with the name. *Follow Active Directory e-mail address* sets userPrincipalName to the value of the Active Directory mail attribute. This option is useful when you want the user's e-mail address to be used for authentication and Active Directory is authoritative for e-mail addresses. *Follow Identity Vault e-mail address* sets userPrincipalName to the value of the Identity Vault e-mail address attribute. This option is useful when you want the user's e-mail address to be used for authentication and the Identity Vault is authoritative for e-mail addresses. *Follow Identity Vault name* is useful when you want to generate userPrincipalName from the user logon name plus a hard-coded string defined in the policy. *None* is useful when you do not want to control userPrincipalName or when you want to implement your own policy.

5.5 Optional Configuration Parameters

The following table explains the parameters that are optional, and that are not prompted for during initial driver configuration.

Table 5-2 Optional Configuration Parameters

Field	Description

Enable DirSync Incremental Values	Implements the LDAP Incremental values control feature. Ordinarily, the publisher will receive all member values of a group when one or more values have changed. This option reports only the added or deleted member values during the poll interval.
	Yes reports only the added or deleted member values during the poll interval.
	No retrieves all member values of a group when one or more values have changed.
	IMPORTANT: This parameter is optional and does not exist when the driver is imported. Add it manually after importing the driver, by adding the following XML code to the driver configuration file:
	`<definition display-name="Enable`
	`DirSync Incremental Values"`
	`hide="false"`
	`id="115"`
	`name="enable-incremental-values"`
	`type="enum">`
	`<description>Ordinarily the publisher`
	`will receive all member values of a`
	`group when one or more has changed.`
	`This option reports only the added or`
	`deleted values during the poll`
	`interval.`
	`Requires 2003 Forest functional mode.`
	`</description>`
	`<enum-choice display-name="Yes">yes</`
	`enum-choice>`
	`<enum-choice display-name="No">no</`
	`enum-choice>`
	`<value>yes</value>`
	`</definition>`
	NOTE: For this feature to work, the Windows 2003 Forest and domain must be in the ‘Windows Server 2003’ functional level.

Field	Description

Overridden LDAP Port Number	Enables you to use a customized LDAP Port number for the LDAP bind.
	Set the options with the following XML code:
	`<definition display-name="Overridden LDAP Port Number" hide="false" id="105" name="ldap-port" type="integer">`
	`        <description>LDAP Port number used for the ldap bind.  Note: Recommended only for binding with an ADAM LDAP server. A 0 (Zero) value implements the default LDAP port numbers. /description>`
	`        <value>0</value>`
	`/definition>`
	NOTE: Configure this field only when you want the driver to bind with an ADAM (Active Directory Application Mode) LDAP server. A value of zero (0) implements the default LDAP port numbers.

Activating the Driver

6

Novell® Identity Manager, Integration Modules, and the Provisioning Module must be activated within 90 days of installation, or they shut down. At any time during the 90 days, or afterward, you can choose to activate Identity Manager products.

To activate the driver, see "Activating Novell Identity Manager Products" in the *Identity Manager 3.5.1 Installation Guide*.

Password Synchronization 7

This section assumes that you are familiar with the information in "Password Synchronization across Connected Systems" in the *Novell Identity Manager 3.5.1 Administration Guide*. The information in this section is specific to this driver.

IMPORTANT: If you have used Password Synchronization 1.0 previously, don't install the new driver shim until you have read "Upgrading Password Synchronization 1.0 to Password Synchronization Provided with Identity Manager" on page 67 and understand the implications. If you install the driver shim, you need to add backward compatibility for Password Synchronization 1.0 to your driver policies at the same time, even if you are not planning to immediately use the Password Synchronization provided with Identity Manager.

In this section:

For information on troubleshooting password synchronization, see "Tips on Password Synchronization" on page 117.

7.1 Comparing Password Synchronization 1.0 and Password Synchronization Provided with Identity Manager

Table 7-1 *Differences Between the Different Versions of Password Synchronization*

Functionality	In Password Synchronization 1.0	In Password Synchronization with Identity Manager
Product delivery	A product separate from Identity Manager.	A feature included with Identity Manager, not sold as a separate product.

Functionality	In Password Synchronization 1.0	In Password Synchronization with Identity Manager
Platforms	◆ Active Directory ◆ NT Domain	Full bidirectional password synchronization is supported on these platforms: ◆ Active Directory ◆ eDirectory™ ◆ NIS ◆ NT Domain These connected systems support publishing user passwords to Identity Manager. Because Universal Password and Distribution Password are reversible, Identity Manager can distribute passwords to connected systems. Any connected system that supports the Subscriber password element can subscribe to passwords from Identity Manager. See "Connected System Support for Password Synchronization" in the *Novell Identity Manager 3.5.1 Administration Guide*.
Password used in eDirectory	eDirectory Password (non-reversible)	Universal Password (reversible), or Distribution Password (also reversible). The eDirectory password can also be kept synchronized, if desired. For example scenarios, see "Implementing Password Synchronization" in the *Novell Identity Manager 3.5.1 Administration Guide*.
Main functionality for Windows connected systems	To provide bidirectional password synchronization so that the eDirectory password is synchronized with the Windows password. However, each workstation requires the Novell® Client™.	To provide bidirectional password synchronization. Because Universal Password and Distribution Password are reversible, passwords can be synchronized in both directions. Accomplished within the Identity Manager Publisher and Subscriber channels.
LDAP password changes	Not supported.	Supported.
Novell Client	Required.	Not required.
nadLoginName attribute	Used for keeping passwords updated.	Not used.

Functionality	In Password Synchronization 1.0	In Password Synchronization with Identity Manager
The component that contains the password synchronization functionality	The Identity Manager driver contained the functionality for updating nadLoginName.	Policies in the driver configuration provide the password synchronization functionality. The driver simply carries out the tasks given by the Metadirectory engine, which come from logic in the policies. The driver manifest, global configuration values, and driver filter settings must also support password synchronization. These are included in the sample driver configurations, or can be added to an existing driver. See Section 7.4, "Upgrading an Existing Driver Configuration to Support Identity Manager Password Synchronization," on page 73.
Agents	A separate piece of software.	No agents are installed; instead, the functionality is now part of the driver.

7.2 Upgrading Password Synchronization 1.0 to Password Synchronization Provided with Identity Manager

If you are currently using Password Synchronization 1.0, complete the instructions in this section to upgrade.

IMPORTANT: Do not install the Identity Manager driver shim until you have reviewed these instructions.

To upgrade from Password Synchronization 1.0 to Password Synchronization provided with Identity Manager:

1 Make sure your environment is ready to use Universal Password.

See "Preparing to Use Identity Manager Password Synchronization and Universal Password" in the *Novell Identity Manager 3.5.1 Administration Guide*.

Enabling Universal Password doesn't automatically cause password changes in both systems. Universal Password synchronization starts working only after users change their passwords.

Scenario: Universal Password. At DigitalAirlines, network administrator Sandy enables Universal Passwords. User Markus logs in and changes his password. The Universal Password for Markus is set on both systems. However, user Marie logs in but doesn't change her password. She continues to log in with her unchanged password. Universal Password functionality for Marie isn't set until she changes her password.

2 Install the Identity Manager 3.5 driver shim to replace the DirXML® 1.1a driver shim, and immediately complete Step 3.

If you are running Identity Manager 2.0, and are using Universal Password, you do not have upgrade Password Synchronization.

Use the installation program as described in the "Installing Identity Manager" section in the *Identity Manager 3.5.1 Installation Guide*, and select only the Identity Manager Driver for Active Directory.

3 Create backward compatibility with Password Synchronization 1.0 by adding a new policy to the driver configuration as described in "Creating Backward Compatibility with Password Synchronization 1.0 by Adding Policies" on page 69.

A DirXML 1.1a driver shim updates the nadLoginName attribute, but the Identity Manager Identity Manager driver shim doesn't. Therefore, you must add policies to the driver configuration to update nadLoginName. This allows Password Synchronization 1.0 to function as usual when you install the driver shim, so no password changes are missed while you finish deploying Identity Manager Password Synchronization.

IMPORTANT: If you don't create backward compatibility, Password Synchronization 1.0 continues to update existing users, but any new or renamed users can't be synchronized until you deploy Identity Manager Password Synchronization.

After you complete this step, you have the Identity Manager 3.5 driver shim and the policies for backward compatibility. Therefore, your driver is supporting Password Synchronization 1.0.

If you can't complete the rest of this procedure right away, you can continue to use Password Synchronization 1.0 until you are ready to finish deploying Identity Manager Password Synchronization.

4 Add support for Identity Manager Password Synchronization to each driver you want to participate in password synchronization.

Either upgrade an existing configuration or replace an existing configuration.

Upgrade an existing configuration: Upgrade your existing DirXML 1.1a driver configuration by converting it to Identity Manager format and adding the policies needed for Identity Manager Password Synchronization:

- Convert the driver to Identity Manager format by using a wizard. See "Upgrading Existing Driver Configurations to Support Password Synchronization" in the *Novell Identity Manager 3.5.1 Administration Guide*.
- Add policies to support Identity Manager Password Synchronization. You can use an "overlay" configuration file to add the policies, driver manifest, and GCVs all at once. You must also add an attribute to the Filter. For instructions, see "Upgrading Existing Driver Configurations to Support Password Synchronization" in the *Novell Identity Manager 3.5.1 Administration Guide*.

Replace the existing configuration with Identity Manager configuration, and add backward compatibility again: The Identity Manager sample driver configuration contains the policies, driver manifest, GCVs, and filter settings to support Identity Manager Password Synchronization. See the instructions in Chapter 5, "Configuring the Active Directory Driver," on page 49 of this driver guide for information on importing the new driver configuration.

- If you choose to replace your existing configuration, make sure you add backward compatibility again, as described in "Creating Backward Compatibility with Password Synchronization 1.0 by Adding Policies" on page 69. The Identity Manager sample driver configuration does not contain those policies.
- Make sure the nadLoginName attribute is set to Publish, as it was in your previous driver configuration.

5 Install new Password Synchronization filters, and configure them if you want the connected system to provide user passwords to Identity Manager.

See Section 7.5, “Setting Up Password Synchronization Filters,” on page 76.

6. Set up SSL, if necessary.

   For instructions, see Section 2.3, “Addressing Security Issues,” on page 23.

   The ability of the driver to set a password in Active Directory (Subscriber channel) requires a secure connection provided by one of the following conditions:

   - The machine running the driver is the same machine as the domain controller.
   - The machine running the driver is in the same domain as the domain controller.
   - The machine not in the domain requires the Simple method and SSL set up between it and the domain controller. Bidirectional password synchronization is available only when using the Negotiate authentication mechanism.

     Refer to Microsoft documentation for instructions, such as Configuring Digital Certificates on Domain Controllers (http://support.microsoft.com/default.aspx?scid=kb;en-us;Q195724).

7. Turn on Universal Password for Identity Vault user accounts by creating Password policies with Universal Password enabled.

   See “Managing Password Synchronization” in the *Identity Manager 3.5.1 Installation Guide*.

   To simplify administration, we recommend that you assign Password policies as high up in the tree as possible.

8. Use the Password Policies and the Password Synchronization settings for the driver to, set up the scenario that you want to use for Password Synchronization.

   See “Implementing Password Synchronization” in the *Novell Identity Manager 3.5.1 Administration Guide*.

9. Test password synchronization by starting the driver and changing a user’s password.

10. After Identity Manager Password Synchronization is working, remove Password Synchronization 1.0.

    10a Using Add/Remove Programs, turn off Password Synchronization 1.0 by removing the agent.

    10b In the filter for the driver, change the nadLoginName attribute to Ignore.

    10c Remove the backward compatibility policies that are updating nadLoginName from the driver configuration.

    10d If desired, you can also remove the nadLoginName attribute from users after Identity Manager Password Synchronization is working, because it is no longer needed.

7.2.1 Creating Backward Compatibility with Password Synchronization 1.0 by Adding Policies

Password Synchronization 1.0 relies on the driver shims updating an attribute named nadLoginName. This attribute indicates whether a user’s password should be synchronized. If a new user was added or the user’s name was changed, the nadLoginName attribute was added or updated to match.

The driver shims in the Identity Manager no longer update this attribute because it is not necessary for Identity Manager Password Synchronization. Therefore, after you install the new driver shim, the nadLoginName attribute is not being updated. This means that Password Synchronization 1.0 no longer receives notice of new or renamed users unless you add backward compatibility to your driver configuration.

For a smooth transition from Password Synchronization 1.0 to Identity Manager Password Synchronization, you need backward compatibility with Password Synchronization 1.0.

For backward compatibility with Password Synchronization 1.0, you must add policies that update the nadLoginName attribute.

These policies must be added regardless of whether you are updating your existing driver configurations, or replacing them with new configurations that ship with Identity Manager. The Identity Manager sample driver configurations for Active Directory do not include the policies by default.

Three policies are necessary, one each for the Subscriber Output Transformation, Publisher Input Transformation, and Publisher Command Transformation. These policies are provided with Identity Manager in a configuration file named Password Synchronization 1.0 Policies for Active Directory. The following procedure explains how to import the new policies and add them to a driver configuration.

1 In iManager, click *Identity Manager Utilities* > *Import Drivers*.

The Import Drivers Wizard opens.

2 Select the driver set where your existing Active Directory driver resides, then click *Next*.

3 In the list of driver configurations that appears, scroll to the *Additional Policies* section and select *Legacy Password Synchronization 1.0 Policies: Backwards Compatibility for AD and NT,* then click *Next*.

4 Complete the import prompts:

4a Select your existing Active Directory driver.

Selecting the existing driver allows you to add the three policies that are necessary. The import process creates three new policy objects, which you must then insert in the appropriate place in the driver configuration.

4b Specify whether the driver is an Active Directory driver.

The imported policies have minor differences depending on which system is chosen.

4c Browse for and select the nadDomain object associated with the driver you want to update.

It can normally be found under the Driver object.

4d (Active Directory only) Specify the name of the eDirectory attribute mapped to the Active Directory attribute sAMAccountName.

You can find this information in the Schema Mapping policy in the driver configuration.

NOTE: If the sAMAccountName is not mapped to any eDirectory attribute, map sAMAccountName to DirXML-ADAlias name.

5 Click *Next*.

Because you chose an existing driver, a page appears asking you to decide how you want the driver to be updated. In this case, you just want to update selected policies.

6 Select *Update Only Selected Policies in That Driver*, and select the check boxes for all three policies listed.

7 Click *Next*, then click *Finish* to complete the wizard.

At this point, the three new policies have been created as Policy objects under the Driver object, but they aren't yet part of the driver configuration. To link them in, you must manually insert each of them at the right point in the driver configuration on the Subscriber and Publisher channels.

8 Insert each of the three new policies into the correct place on your existing driver configuration.

If any of these parts of the driver configuration has multiple policies, make sure these new policies are listed last.

Table 7-2 *Policies*

Policy Object Name	Where To Insert It
PassSync(Pub)-Command Transform Policies	Command Transformation Policies on the Publisher channel.
PassSync(Pub)-Input Transform Policies	Input Transformation Policies on the Publisher channel.
PassSync(Sub)-Output Transform Policies	Output Transformation Policies on the Subscriber channel.

Repeat Step 8a through Step 8f for each policy.

8a Click *Identity Manager > Identity Manager Overview*.

8b Select the driver set for the driver you are updating.

8c Click the driver you just updated.

A page opens, showing a graphical representation of the driver configuration.

8d Click the icon for the place where you need to add one of the three new policies.

8e Click *Insert* to add the new policy.

On the Insert page that appears, click *Use an Existing Policy*, browse for the new policy object, then click *OK*.

8f If you have more than one policy in the list for any of the three new policies, use the arrow buttons to move the new policy down so it is last in the list.

9 Repeat Step 1 though Step 8 for all your Active Directory drivers.

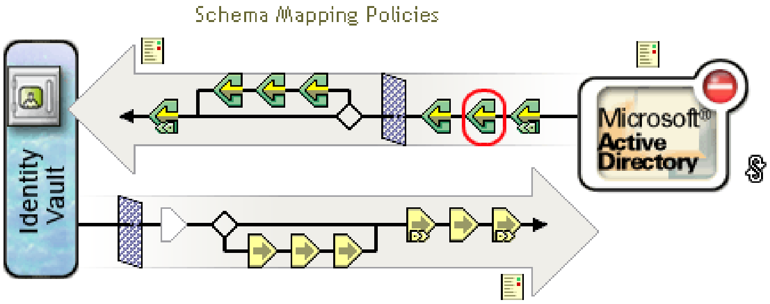
If the sAMAccountName needs to be mapped to the DirXML-ADAliasName in the Publisher channel Schema Mapping policy, follow this procedure:

WARNING: If the sAMAccountName is mapped to another attribute, following this procedure invalidates your policies. The policies stop synchronizing passwords. Make sure you provide the proper attribute in Step 4d on page 70.

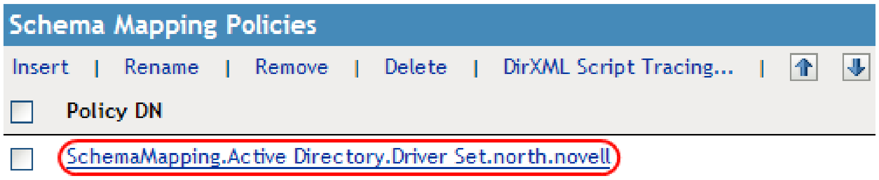
1 In iManager, select *Identity Manager > Identity Manager Overview*.

2 Browse to and select the Driver Set object that contains the Active Directory driver, then click *Search*.

3 Click the driver icon, then click the *Schema Mapping Policies* icon for the Publisher channel.



4 Click the policy name to edit it.

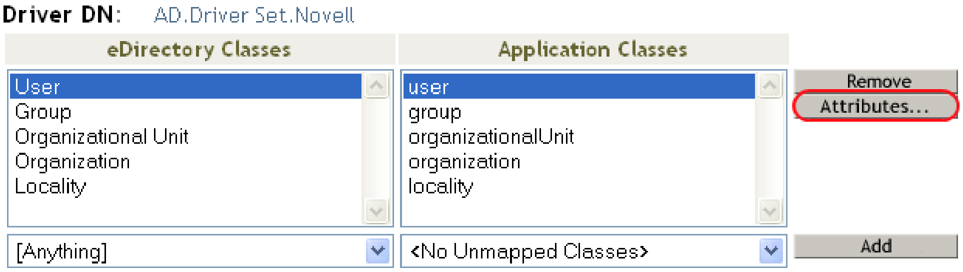


Schema Mapping Policies

Insert | Rename | Remove | Delete | DirXML Script Tracing... |

□	Policy DN
□	SchemaMapping.Active Directory.Driver Set.north.novell

5 Select the User class, then click *Attributes*.

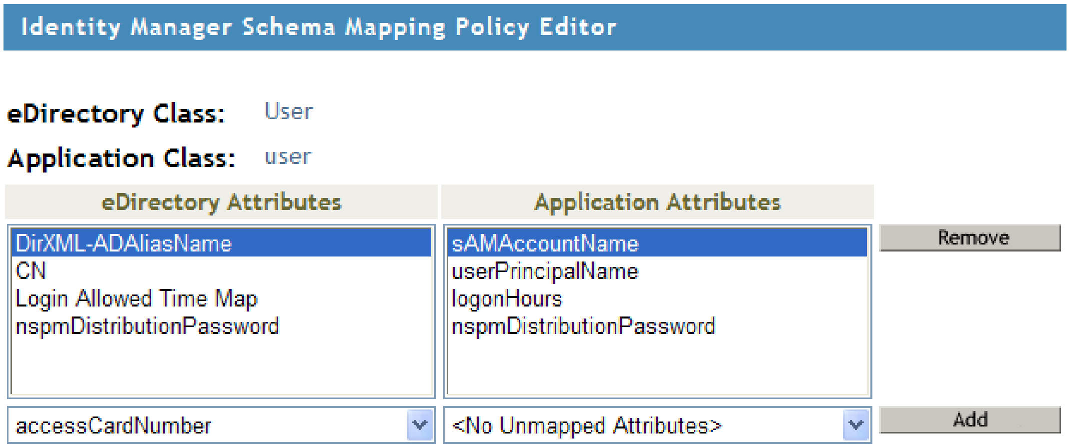


Driver DN: AD.Driver Set.Novell

eDirectory Classes	Application Classes
User	user
Group	group
Organizational Unit	organizationalUnit
Organization	organization
Locality	locality
[Anything]	<No Unmapped Classes>

6 Click the drop-down list under *eDirectory Attributes*, then browse to and select *DirXML-ADAliasName*.

7 Click the drop-down list under *Application Attributes*, then browse to and select *sAMAccountName*.



Identity Manager Schema Mapping Policy Editor

eDirectory Class: User

Application Class: user

eDirectory Attributes	Application Attributes
DirXML-ADAliasName	sAMAccountName
CN	userPrincipalName
Login Allowed Time Map	logonHours
nspmDistributionPassword	nspmDistributionPassword
accessCardNumber	<No Unmapped Attributes>

8. Click *Add*, then click *OK*.
9. Select the Group class, then click *Attributes*.
10. Repeat Step 6 through Step 8 for the Group class.
11. Click *OK* twice.

After you have completed this procedure, the driver configurations for your Active Directory drivers are backward compatible with Password Synchronization 1.0. This means that Password Synchronization continues to function as it did before, allowing you to upgrade to Identity Manager Password Synchronization at your convenience.

7.3 New Driver Configuration and Identity Manager Password Synchronization

If you are not using Password Synchronization 1.0, and you are creating a new driver or replacing an existing driver's configuration with the Identity Manager configuration, follow the instructions in "Configuring and Synchronizing a New Driver " in the *Novell Identity Manager 3.5.1 Administration Guide*.

In addition, do the following:

- Set up SSL, if necessary. See Section 2.3, "Addressing Security Issues," on page 23.

  The ability of the driver to set a password in Active Directory (Subscriber channel) requires a secure connection provided by one of the following conditions:

  - The machine running the driver is the same machine as the domain controller.
  - The machine running the driver is in the same domain as the domain controller.
  - The machine not in the domain requires the Simple method and SSL set up between it and the domain controller. Bidirectional password synchronization is available only when using the Negotiate authentication mechanism.

    Refer to Microsoft documentation for instructions, such as Enabling Secure Sockets Layer for SharePoint Portal Server 2003 (http://office.microsoft.com/en-us/assistance/HA011648191033.aspx).
- Install new Password Synchronization filters and configure them if you want the connected system to provide user passwords to Identity Manager. See Section 7.5, "Setting Up Password Synchronization Filters," on page 76.
- Set up the scenario for Password Synchronization that you want to use, using the Password Policies and the Password Synchronization settings for the driver. See "Implementing Password Synchronization" in the *Novell Identity Manager 3.5.1 Administration Guide*.

7.4 Upgrading an Existing Driver Configuration to Support Identity Manager Password Synchronization

IMPORTANT: If a driver is being used with Password Synchronization 1.0, you should complete this section only with Section 7.2, "Upgrading Password Synchronization 1.0 to Password Synchronization Provided with Identity Manager," on page 67, not alone.

The following is an overview of the tasks you use the procedure in this section to complete:

- Add driver manifest, global configuration values, and password synchronization policies to the driver configuration. For a list of the policies you add, see “Policies Required in the Driver Configuration” in the *Novell Identity Manager 3.5.1 Administration Guide*.
- Change the filter to allow Subscriber notify and Publisher ignore on the nspmDistributionPassword attribute.

Prerequisites

□ Make sure you have converted your existing driver to Identity Manager format, as described in “Upgrading a Driver Configuration from DirXML 1.1a to Identity Manager 3.5.1 Format” in the *Novell Identity Manager 3.5.1 Administration Guide*.

□ Use the Export Drivers Wizard to create a backup of your existing driver.

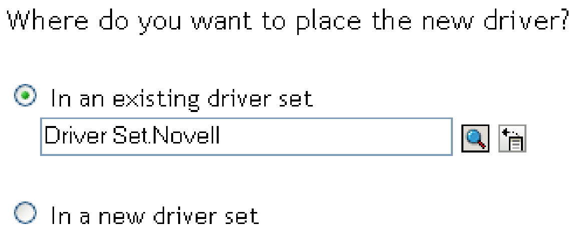
□ Make sure you have installed the new driver shim. Some password synchronization features such as Check Password Status won’t work without the Identity Manager driver shim.

Procedure

1 In iManager, click *Identity Manager Utilities* > *Import Configurations*.

The Import Drivers Wizard opens.

2 Select the driver set where your existing driver resides, then click *Next*.



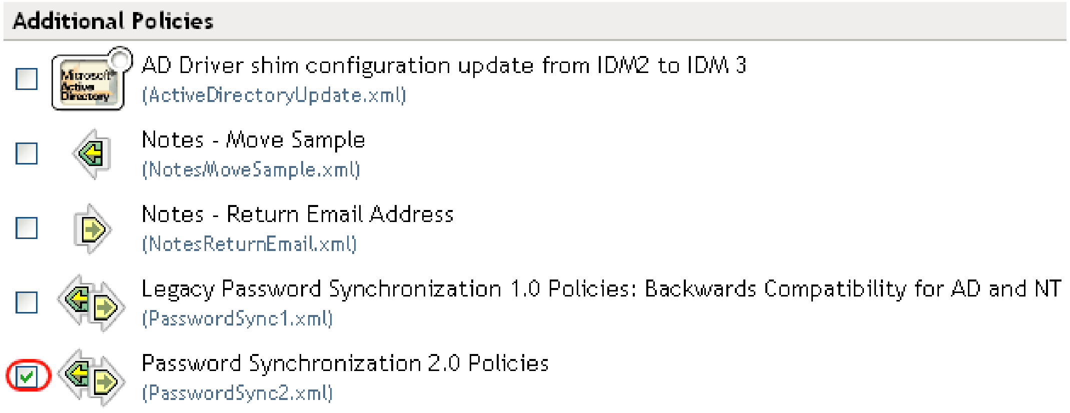
Where do you want to place the new driver?

● In an existing driver set

Driver Set.Novell

○ In a new driver set

3 In the list of driver configurations that appears, select *Password Synchronization 2.0 Policies*, then click *Next*.



Additional Policies

□ AD Driver shim configuration update from IDM2 to IDM 3 (ActiveDirectoryUpdate.xml)

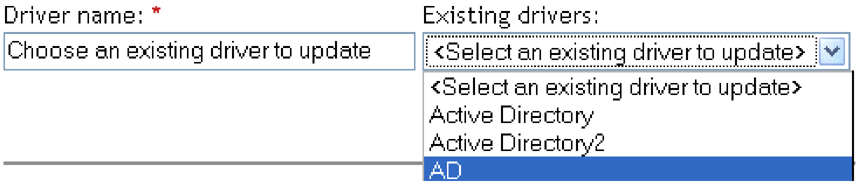
□ Notes - Move Sample (NotesMoveSample.xml)

□ Notes - Return Email Address (NotesReturnEmail.xml)

□ Legacy Password Synchronization 1.0 Policies: Backwards Compatibility for AD and NT (PasswordSync1.xml)

☑ Password Synchronization 2.0 Policies (PasswordSync2.xml)

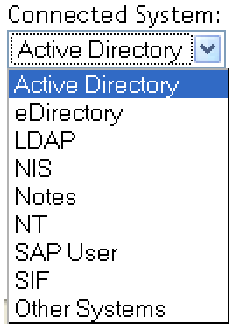
4 Select the name of the Active Directory driver to update from the drop-down list.



Driver name: * Choose an existing driver to update

Existing drivers: <Select an existing driver to update>

5 Select *Active Directory* as the connected system, then click *Next*.



Connected System: Active Directory

6 Select *Update everything about that driver and policy libraries*, then click *Next*.

This option gives you the driver manifest, global configuration values (GCVs), and password policies necessary for password synchronization.

The driver manifest and GCVs overwrite any values that already exist. Make sure you have record any existing GCVs before updating.

The password policies don't overwrite any existing policy objects. They are simply added to the Driver object.

If you do have driver manifest or GCV values that you want to save, choose the option named *Update only Selected Policies* for that driver, and select the check boxes for all the policies. This option imports the password policies but doesn't change the driver manifest or GCVs.

7 Click *Next*, then click *Finish* to complete the wizard.

At this point, the new policies have been created as policy objects under the driver object. However, the new policies aren't yet part of the driver configuration. To link them in, you must manually insert each of them at the right point in the driver configuration on the Subscriber and Publisher channels.

8 Insert each of the new policies into the correct place in your existing driver configuration.

If a policy set has multiple policies, make sure these password synchronization policies are listed last.

The list of the policies and where to insert them is in "Policies Required in the Driver Configuration" in the *Novell Identity Manager 3.5.1 Administration Guide*.

Repeat Step 8a through Step 8e for each policy.

8a Click *Identity Manager > Identity Manager Overview*, then select the driver set for the driver you are updating.

8b Click the driver you just updated.

A page opens, showing a graphical representation of the driver configuration.

8c Click the icon for the place where you need to add one of the new policies.

8d Click *Insert* to add the new policy.

On the Insert page that appears, click *Use an Existing Policy*, browse for the new policy object, then click *OK*.

8e If you have more than one policy in the list for any of the new policies, use the arrow buttons to move the new policies to the correct location in the list.

Make sure the policies are in the order listed in "Policies Required in the Driver Configuration" in the *Novell Identity Manager 3.5.1 Administration Guide*.

9 Change the filter for the driver to allow the nspmDistributionPassword attribute to be synchronized.

Enable *Notify* only on the Subscriber channel. Set the Publisher channel to *Ignore*.

10 Set up SSL, if necessary.

Instructions are contained in Section 2.3, "Addressing Security Issues," on page 23.

The ability of the driver to set a password in Active Directory (Subscriber channel) requires a secure connection provided by one of the following conditions:

- The machine running the driver is the same machine as the domain controller.
- The machine running the driver is in the same domain as the domain controller.
- The machine not in the domain requires the Simple method and SSL set up between it and the domain controller. Bidirectional password synchronization is available only when using the Negotiate authentication mechanism.

  Refer to Microsoft documentation for instructions, such as Configuring Digital Certificates on Domain Controllers (http://www.microsoft.com/technet/security/prodtech/windows2000/secwin2k/swin2kad.mspx).

11 Install new Password Synchronization filters and configure them if you want the connected system to provide user passwords to Identity Manager. See Section 7.5, "Setting Up Password Synchronization Filters," on page 76.

At this point, the driver has the new driver shim, Identity Manager format, and the other pieces that are necessary to support password synchronization: driver manifest, GCVs, password synchronization policies, and filters. Now you can specify how you want passwords to flow to and from connected systems, using the Password Synchronization interface in iManager.

12 Set up the scenario for Password Synchronization that you want to use, using the Password Policies and the Password Synchronization settings for the driver.

See "Implementing Password Synchronization" in the *Novell Identity Manager 3.5.1 Administration Guide*.

13 Repeat Step 1through Step 12 for all the drivers that you want to participate in password synchronization.

7.5 Setting Up Password Synchronization Filters

The Active Directory driver must be configured to run on only one Windows machine. However, for password synchronization to occur, you must install a password filter (`pwFilter.dll`) on each domain controller and configure the registry to capture passwords to send to the Identity Vault.

The password filter is automatically started when the domain controller is started. The filter captures password changes that users make by using Windows clients, encrypts the changes, and sends them to the driver to update the Identity Vault. For more information about configuring Password Synchronization, see "Implementing Password Synchronization" in the *Novell Identity Manager 3.5.1 Administration Guide*.

To simplify setup and administration of password filters, an Identity Manager PassSync utility is added to the Control Panel when the driver is installed. This utility gives you two choices for setting up the password filters, depending on whether you are willing to allow remote access to the registry on your domain controllers:

- Section 7.5.1, "Allowing Remote Access to the Registry," on page 77
- Section 7.5.2, "Not Allowing Remote Access to the Registry," on page 81

7.5.1 Allowing Remote Access to the Registry

If you allow remote access to the registry of each domain controller from the machine where you are running the driver, use the procedure in this section to configure the password filter. It allows the Identity Manager PassSync utility to configure each domain controller from one machine.

If you configure all the domain controllers from one machine, the Identity Manager PassSync utility provides the following features to help you during setup:

- Lets you specify which domain you want to participate in password synchronization.
- Automatically discovers all the domain controllers for the domain.
- Lets you remotely install the `pwFilter.dll` on each domain controller.
- Automatically updates the registry on the machine where the driver is running and on each domain controller.
- Lets you view the status of the filter on each domain controller.
- Lets you reboot a domain controller remotely.

  Rebooting the domain controller is necessary when you first add a domain for password synchronization, because the filter that captures password changes is a DLL file that starts when the domain controller is started.

Because setting up the filter requires rebooting the domain controller, you might want to perform this procedure after hours, or reboot only one domain controller at a time. If the domain has more than one domain controller, keep in mind that each domain controller where you want Password Synchronization to function must have the filter installed and must be rebooted.

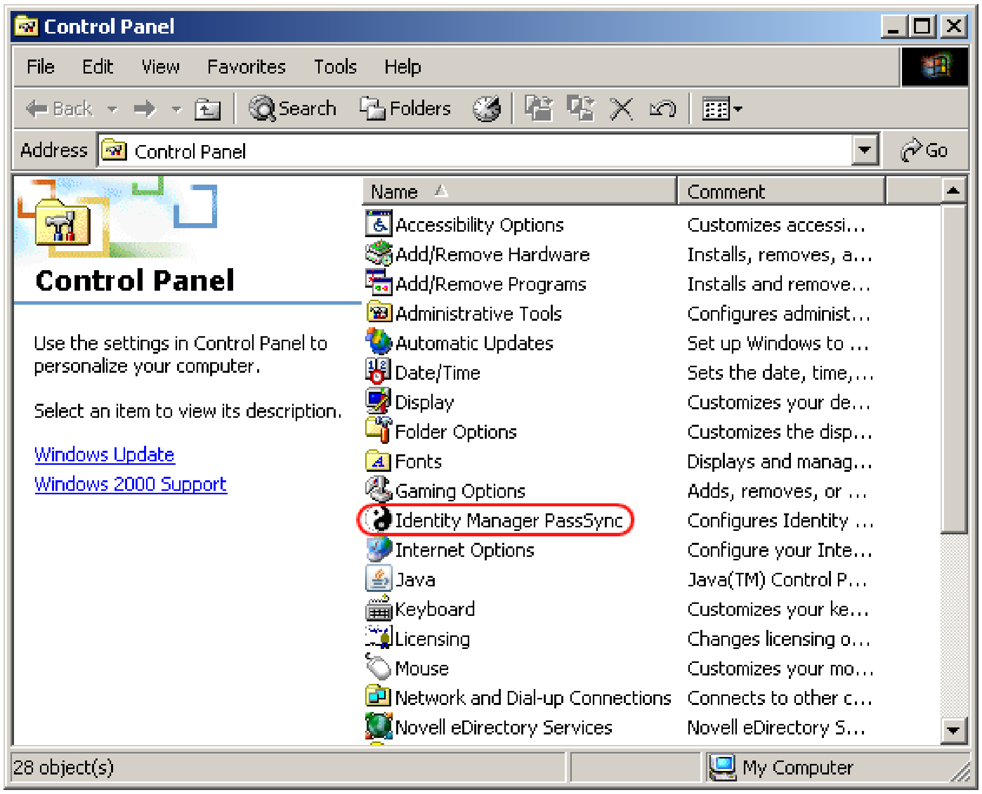
1. Confirm that port 135 (the RPC endpoint mapper) is accessible on the domain controllers and on the machine where the Identity Manager Driver for Active Directory is configured to run.

   If you are using NetBIOS over TCP, you also need these ports:

   - 137: NetBIOS name service
   - 138: NetBIOS datagram service
   - 139: NetBIOS session service

   A firewall could prevent the ports from being accessible remotely.
2. Log in with an administrator account on the computer where the driver is installed.

3 At the computer where the driver is installed, click *Start* > *Control Panel* > *Identity Manager PassSync*.

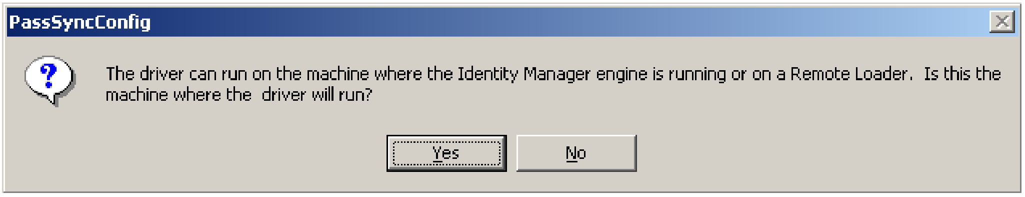


Use the settings in Control Panel to personalize your computer.

Select an item to view its description.

Name	Comment
Accessibility Options	Customizes accessi...
Add/Remove Hardware	Installs, removes, a...
Add/Remove Programs	Installs and remove...
Administrative Tools	Configures administ...
Automatic Updates	Set up Windows to ...
Date/Time	Sets the date, time,...
Display	Customizes your de...
Folder Options	Customizes the disp...
Fonts	Displays and manag...
Gaming Options	Adds, removes, or ...
Identity Manager PassSync	Configures Identity ...
Internet Options	Configure your Inte...
Java	Java(TM) Control P...
Keyboard	Customizes your ke...
Licensing	Changes licensing o...
Mouse	Customizes your mo...
Network and Dial-up Connections	Connects to other c...
Novell eDirectory Services	Novell eDirectory S...

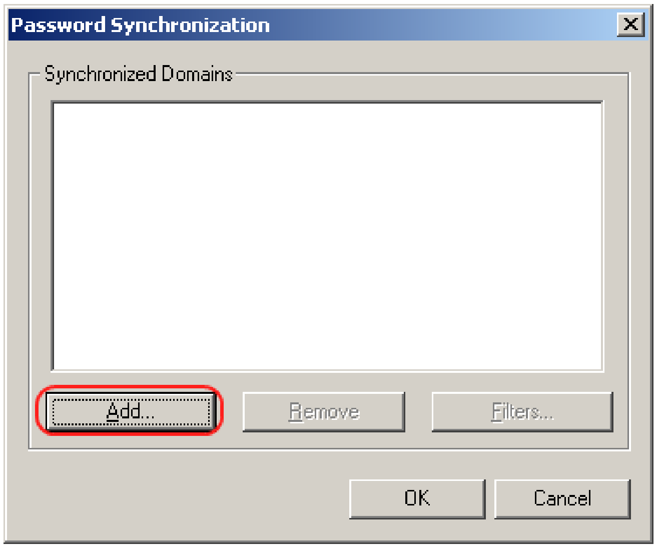
4 In the dialog box that is displayed, click *Yes* to specify that is the machine where the driver is installed.



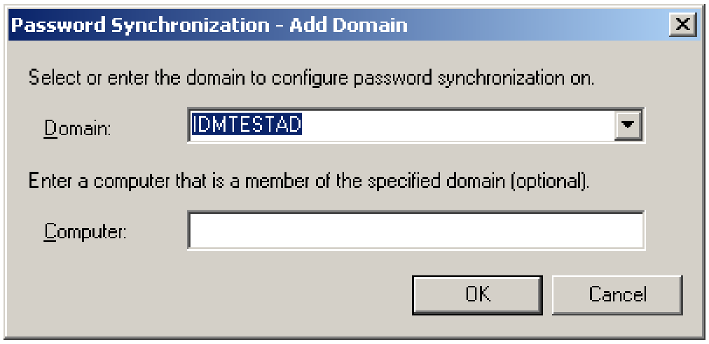
The driver can run on the machine where the Identity Manager engine is running or on a Remote Loader. Is this the machine where the driver will run?

The first time you open the utility, it asks whether this is the machine where the Identity Manager driver is installed. After you complete the configuration, you are not shown this prompt again unless you remove this domain from the list.

5 Click *Add*, then browse to and select the domain that you want to participate in password synchronization.



The drop-down list displays known domains.


Password Synchronization - Add Domain

Select or enter the domain to configure password synchronization on.

Domain: IDMTESTAD

Enter a computer that is a member of the specified domain (optional).

Computer:

The Identity Manager PassSync utility discovers all the domain controllers for that domain, and installs `pwFilter.dll` on each domain controller. It also updates the registry on the computer where you are running the drivers, and on each domain controller. This might take a few minutes.

The `pwFilter.dll` doesn’t capture password changes until the domain controller has been rebooted. The Identity Manager PassSync utility lets you see a list of all the domain controllers and the status of the filter on them. It also lets you reboot the domain controller from inside the utility.

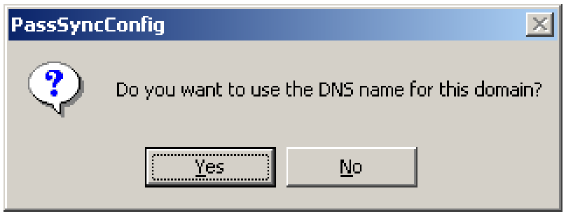
6 (Optional) Specify a computer in the domain, then click *OK*.

If you leave the *Computer* edit box blank, PassSync queries the local machine. Therefore, if you are running PassSync on a domain controller, you don’t need to specify a name. PassSync queries the local machine (in this case, a domain controller) and gets (from the database) the list of all domain controllers in the domain.

If you aren’t installing on a domain controller, specify the name of a computer that is in the domain and that can get to a domain controller.

If you receive an error message indicating that PassSync can’t locate a domain, specify a name.

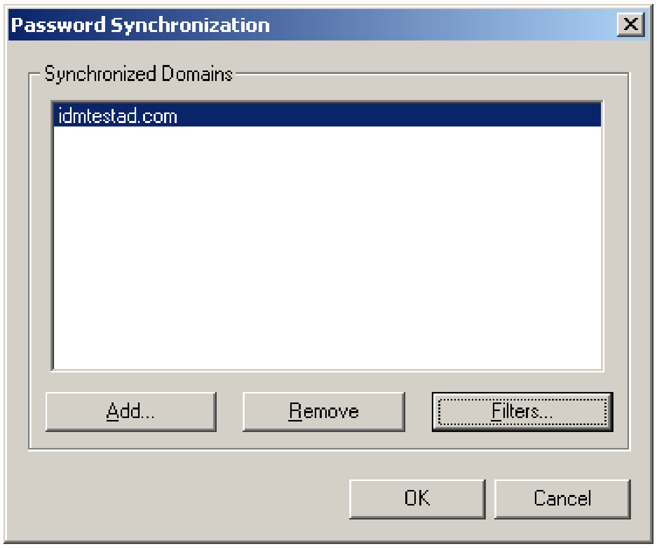
7 Click *Yes* to use the domain’s DNS name.


PassSyncConfig

Do you want to use the DNS name for this domain?

You can select *No*, but the DNS name provides more advanced authentication and the ability to more reliably discover domains in bigger installations. However, the choice depends on your environment.

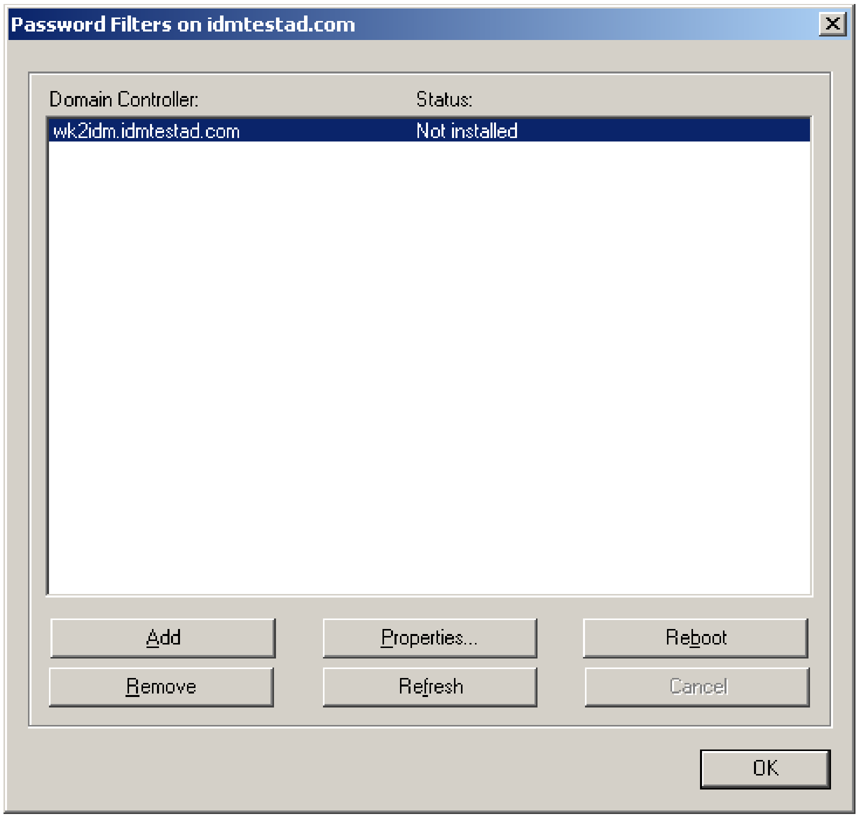
8 Select the name of the domain you want to participate in password synchronization from the list, then click *Filters*.


Password Synchronization

Synchronized Domains

idmtestad.com

The utility displays the names of all the domain controllers in the selected domain and the status of the filter.



Domain Controller:	Status:
wk2idm.idmtestad.com	Not installed

The status for each domain controller should display the filter state as *Not installed*. However, it might take a few minutes for the utility to complete its automated task, and in the meantime the status might say *Unknown*.

9 To install the filter, click *Add*, then click *Reboot*.

You can choose to reboot the domain controllers at a time that makes sense for your environment. Just keep in mind that password synchronization won't be fully functional until every domain controller has been rebooted.

10 When the status for all domain controllers is *Running*, test password synchronization to confirm that it is working. For more information on how to test password synchronization, see "Implementing Password Synchronization" in the *Novell Identity Manager 3.5.1 Administration Guide*.

11 To add more domains, click *OK* to return to the list of domains, and repeat Step 5 through Step 10.

7.5.2 Not Allowing Remote Access to the Registry

If you do not want to allow remote access to the registry of each domain controller, you must set up the password filters on each domain controller separately. To do this, go to each domain controller, install the driver files so you have the Identity Manager PassSync utility, and use the utility on each machine to install the password filter and update the registry.

In this procedure, you install the driver so that you have the Identity Manager PassSync utility. Then you use the utility to install the `pwFilter.dll` file, specify the port to use, and specify which host machine is running the Identity Manager Driver for Active Directory.

Because setting up the filter requires rebooting the domain controller, you might want to perform this procedure after hours, or reboot only one domain controller at a time. If a domain has more than one domain controller, keep in mind that each domain controller where you want Password Synchronization to function must have the filter installed and must be rebooted.

This procedure is for any Domain Control that does not have the Active Directory driver installed on it.

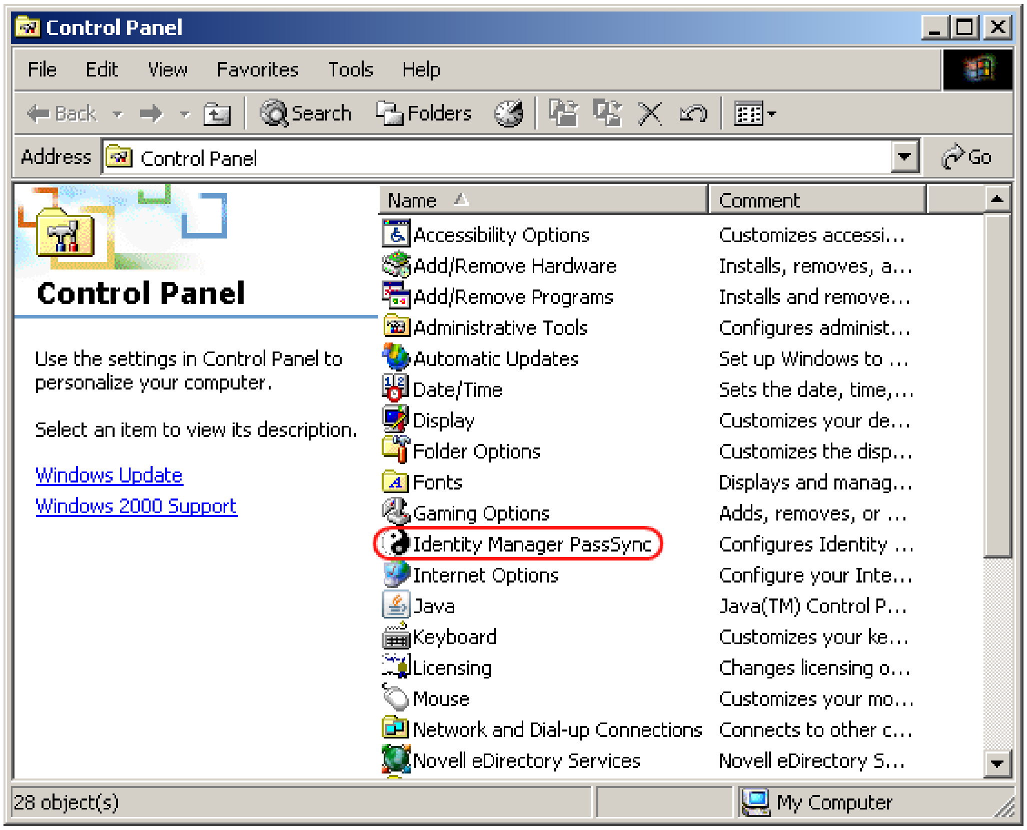
1 Confirm that the following ports are available on both the domain controller and the machine where the Identity Manager Driver for Active Directory is configured to run:

- 135: The RPC endpoint mapper
- 137: NetBIOS name service
- 138: NetBIOS datagram service
- 139: NetBIOS session service

2 On the domain controller, use the Identity Manager Installation to install only the Identity Manager Driver for Active Directory.

Installing the driver installs the Identity Manager PassSync utility.

3 Click *Start* > *Settings* > *Control Panel* > *Identity Manager PassSync.*



Address: Control Panel

Control Panel

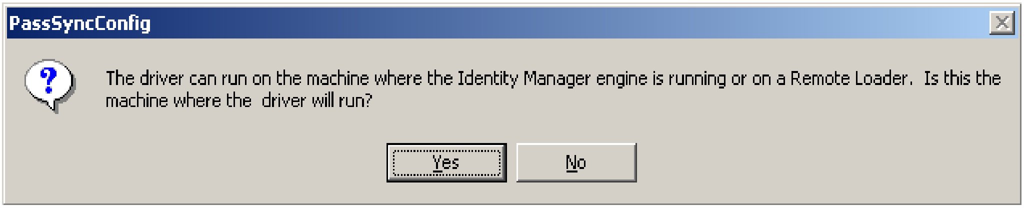
Use the settings in Control Panel to personalize your computer.

Select an item to view its description.

Windows Update
Windows 2000 Support

Name	Comment
Accessibility Options	Customizes accessi...
Add/Remove Hardware	Installs, removes, a...
Add/Remove Programs	Installs and remove...
Administrative Tools	Configures administ...
Automatic Updates	Set up Windows to ...
Date/Time	Sets the date, time,...
Display	Customizes your de...
Folder Options	Customizes the disp...
Fonts	Displays and manag...
Gaming Options	Adds, removes, or ...
Identity Manager PassSync	Configures Identity ...
Internet Options	Configure your Inte...
Java	Java(TM) Control P...
Keyboard	Customizes your ke...
Licensing	Changes licensing o...
Mouse	Customizes your mo...
Network and Dial-up Connections	Connects to other c...
Novell eDirectory Services	Novell eDirectory S...

4 In the dialog box that displays, click *No* to specify that this machine is not running the Active Directory driver.

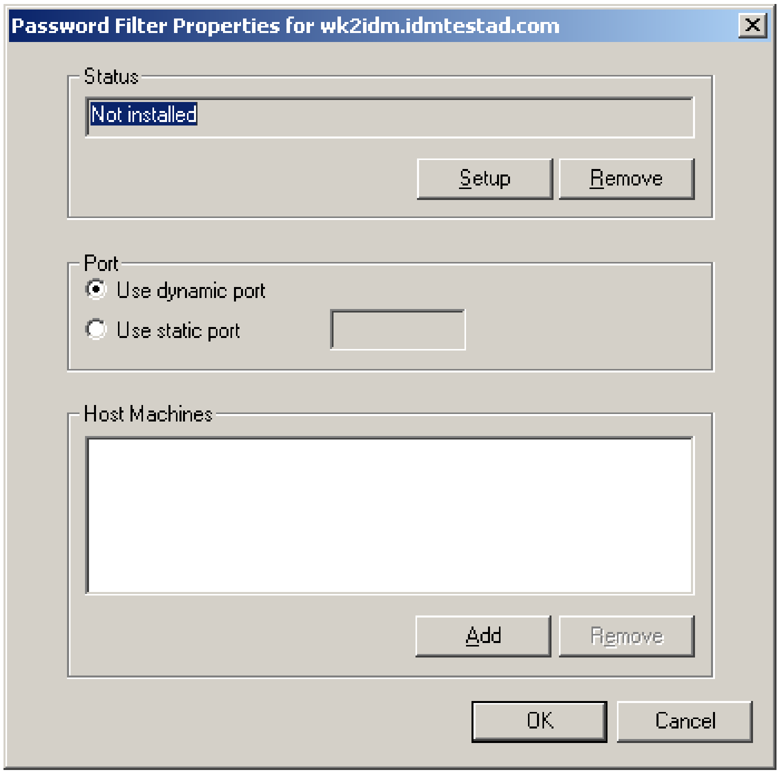


PassSyncConfig

The driver can run on the machine where the Identity Manager engine is running or on a Remote Loader. Is this the machine where the driver will run?

After you complete the configuration, you are not shown this prompt again unless you remove the password filter by using the Remove button in the Password Filter Properties dialog box.

5 After you click *No*, the Password Filter Properties dialog box appears, with a status message indicating that the password filter is not installed on this domain controller.



Password Filter Properties for wk2idm.idmtestad.com

Status: Not installed

Port
● Use dynamic port
○ Use static port

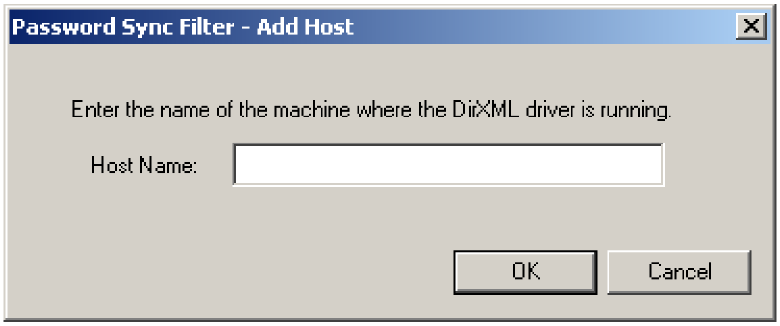
Host Machines

6 Click the *Setup* button to install the password filter, `pwFilter.dll`.

7 For the *Port* setting, specify whether to use dynamic port or static port.

Use the static port option only if you have decided to configure your remote procedure call (RPC) for the domain controller differently than the default.

8 Click *Add* to specify the Host Name of the machine running the Identity Manager driver, then click *OK*.


Password Sync Filter - Add Host

Enter the name of the machine where the DirXML driver is running.

Host Name:

This step is necessary so that the password filter knows where to send the password changes. The password filter captures password changes, and must send them to the Identity Manager driver to update the Identity Manager data store.

9. Verify that the information specified in Step 6 through Step 8 is correct, then click *OK*.
10. Reboot the domain controller to complete the installation of the password filter.

    You can choose to reboot at a time that makes sense for your environment. Just keep in mind that password synchronization won't be fully functional until every domain controller has the password filter installed and has been rebooted.

    After the installation is complete and the domain controller is rebooted, the password filter is loaded automatically whenever the domain controller starts up.
11. Check the status for the password filter again by clicking *Start* > *Settings* > *Control Panel*, and double-clicking the Identity Manager PassSync utility.

    Confirm that the status says Running.
12. Repeat Step 2 through Step 11 for each domain controller that you want to participate in Password Synchronization.
13. When the status says Running for all the domain controllers, test Password Synchronization to confirm that it is working. For more information on how to test password synchronization, see "Implementing Password Synchronization" in the *Novell Identity Manager 3.5.1 Administration Guide*.

7.6 Retrying Synchronization after a Failure

The driver and the password filter have been enhanced to improve how password synchronization is retried after a failure.

- Section 7.6.1, "Retrying after an Add or Modify Event," on page 84
- Section 7.6.2, "Password Expiration Time," on page 85

7.6.1 Retrying after an Add or Modify Event

If a password change sent from Active Directory is not completed successfully in the Identity Vault, the driver caches the password. It is not retried again until an Add or Modify event occurs for the user that the password belongs to. (Previously, these saved passwords were retried at every polling interval.)

When the driver polls for changes in Active Directory, the driver receives Add or Modify events for users. For each user Add or Modify event, the driver checks to see if it has a password saved for this new user. If it does, the driver sends the password to the Identity Vault as a Modify user event.

If you have set up Password Synchronization to send e-mail messages to users when password synchronization fails, this enhancement minimizes the number of e-mails that a user might receive.

7.6.2 Password Expiration Time

A parameter named Password Expiration Time has been added. This parameter lets you determine how long to save a particular user's password if synchronization is not successful on the first try. The driver saves a password until it is successfully changed in the Identity Vault, or until the Password Expiration Time elapses.

You are prompted to specify an expiration time when you import the sample driver configuration. If you don't specify a time, or if the interval field contains invalid characters, the default setting is 60 minutes. If the time specified is less than three times the polling interval specified, the driver changes the time to be three times the polling interval.

Set the value large enough to handle whatever temporary backlog of passwords exists. If you are doing bulk changes, set the timeout large enough to handle all the changes. The rule of thumb is to allow one second per password. For example, to synchronize 18,000 passwords, allow 300 minutes (18,000 passwords divided by 60 seconds).

A setting of -1 is indefinite. Although this setting can handle bulk changes, it can cause problems. For example, a password might never be synchronized because the account wasn't associated. Such a password would therefore remain in the system forever. A number of similar situations could result in a large inventory of unsynchronized passwords held by the system.

Scenarios Relating to Password Expiration Time

On the Publisher channel, password synchronization might occur before the Add event. The driver retries immediately following the Add event.

- "Scenario: No Effect" on page 85
- "Scenario: Increasing the Expiration Time" on page 86
- "Scenario: Never Meeting Requirements" on page 86
- "Scenario: E-Mail Notifications" on page 87

Scenario: No Effect

A new user with a password is created in Active Directory. The filter immediately sends the new password to the driver. However, the driver hasn't yet received that user Add event because the event occurred between polling intervals. Because the driver has not yet created the user in the Identity Vault, the password synchronization is not successful on this first attempt. The driver caches the password.

At the next polling interval, the driver receives the Add user event for the new user. The driver also checks to see if it has a password cached for this new user. The driver sends the Add user event to the Identity Vault, and also sends a Modify user event to synchronize the password.

In this case, the password synchronization is delayed by only one polling interval.

The Password Expiration Time parameter doesn't have an effect in this situation.

Scenario: Increasing the Expiration Time

A new user with a password is created in Active Directory. However, the user information doesn't meet the requirements of the Create policy for the Active Directory driver.

For example, perhaps the Create rule requires a full name, and the required information is missing. Like the No Effect example, the filter immediately sends the password change to the driver. However, on the first try the password change is not successful in the Identity Vault because the user doesn't exist yet. The driver caches the password.

In this case, however, even when the driver polls for changes in Active Directory and discovers the new user, the driver can't create the new user because the user information doesn't meet the Create policy's requirements.

Creating the new user and synchronizing the password are delayed until all the user information is added in Active Directory to satisfy the Create policy. Then the driver adds the new user in the Identity Vault, checks to see if it has a password cached for this new user, and sends a Modify user event to synchronize the password.

The Password Expiration Time parameter affects this scenario only if the time interval elapses before the user information in Active Directory meets the requirements of the Create policy. If the Add event comes in after the password has expired and the driver doesn't have the password cached for that user, synchronization can't occur. Because the driver doesn't have a cached password, the driver uses the default password in the password policy.

After the user changes the password in either Active Directory or the Identity Vault, that password is synchronized.

If Password Synchronization is set up for bidirectional flow of passwords, a password can also be synchronized from the Identity Vault to Active Directory when a password change is made in the Identity Vault.

If your Create policy is restrictive, and it generally takes longer than a day for a new user's information to be completed in Active Directory, you might want to increase the Password Expiration Time parameter interval accordingly. The driver can then cache the passwords until the user is finally created in the Identity Vault.

Scenario: Never Meeting Requirements

A user with a password is created in Active Directory. However, this user never meets the criteria of the Create policy for the Active Directory driver.

For example, perhaps the new user in Active Directory has a Description that indicates the user is a contractor, and the Create policy blocks creation of User objects for contractors because the business policy is that contract employees are not intended to have a corresponding user account in the Identity Vault. Like the previous example, the filter immediately sends the password change, but the password synchronization isn't successful on the first attempt. The driver caches the password.

In this case, a corresponding user account is never created in the Identity Vault. Therefore, the driver never synchronizes the cached password. After the Password Expiration Time has passed, the driver removes the user password from its cache.

Scenario: E-Mail Notifications

Markus has an Active Directory account and a corresponding Identity Vault account. He changes his Active Directory password, which contains six characters. However, the password doesn't meet the eight-character minimum required by the Password Policy that the administrator created in eDirectory. Password Synchronization is configured to reject passwords that do not meet the policy and to send a notification e-mail to Markus saying that password synchronization failed. The driver caches the password and retries it only if a change is made to the User object in Active Directory.

In this case, shortly after changing a password, Markus receives an e-mail stating that the password synchronization wasn't successful. Markus receives the same e-mail message each time the driver retries the password.

If Markus changes the password in Active Directory to one that complies with the Password Policy, the driver synchronizes the new password to the Identity Vault successfully.

If Markus doesn't change to a compliant password, the password synchronization is never successful. When the Password Expiration Time elapses, the driver deletes the cached password and no longer retries it.

7.7 Synchronizing Passwords on the Subscriber Channel Requires a Secure Connection

For any password synchronization operation to synchronize on the Subscriber channel, the driver must have SSL enabled or have signing and sealing enabled. If the connection is not secure, the password operation is not synchronized. Enabling SSL or signing and sealing is done in the driver parameters. For more information, see Section E.1.5, "Driver Parameters," on page 171.

The different operations could be a password modify, a password check, or generating a new password. For more information, see Section 2.3.2, "Encryption," on page 24.

7.8 Disabling Password Synchronization on a Driver

You can disable password synchronization on a driver by setting the *Password Sync Timeout* parameter to 0. Sometimes there is a need to have two Active Directory drivers enabled for one domain, but you only want one driver handling the password synchronization. Make sure that the *Password Sync Timeout* parameter is set to 0 on the driver that does not synchronize passwords.

A use case for this is if one driver is synchronizing User objects and another driver is synchronizing Contacts. Contacts are displayed in the Exchange Global Address List (GAL), but they do not require an Active Directory license because they do not authenticate.

See "Password Sync Timeout (minutes)" on page 173 in the Appendix E, "Properties of the Driver," on page 167 for more information about this parameter.

Managing the Driver

8

The driver can be managed through Designer, iManager, or the DirXML® Command Line utility.

8.1 Managing Groups

The Active Directory group class defines two types of groups and three scopes for membership in the group. Type and scope are controlled by the groupType attribute, which can be set via an Identity Manager policy when a group is created in Active Directory and changed by modifying the attribute.

A group holds a collection of object references. The Distribution Group type gives no special rights or privileges to its members and is commonly used as a distribution list for Exchange. The Security Group type is a security principal. Its members receive the rights and privileges of the group. Security Groups have a pre-Windows 2000 logon name (samAccountName) and a Security Identifier (SID) that can be used in Security Descriptor (SD) Access Control Lists (ACL) on other objects to grant or deny rights and privileges to its members.

Group scope controls whether an object from a foreign domain can be a member of the group and also whether the group itself can be a member of another group. The three scopes are Domain Local, Global, and Universal. How these scopes behave, or whether the scope is valid at all, depends on whether Active Directory is operating in Windows 2000 Mixed, Windows 2000 Native, or Windows 2003 mode.

In general, Domain Local groups can hold references to objects anywhere in the forest but can be assigned permissions only within the domain. Global groups are the opposite. They can only hold references to objects within the domain but can be assigned permissions throughout the forest. Universal groups can hold references and can be assigned permissions throughout the forest. However, Universal groups come with their own restrictions and performance issues. Groups should be created and used in conformance with Microsoft recommendations.

The groupType attribute is a 32-bit integer whose bits define type and scope. Groups can have only a single scope at any given time.

Table 8-1 *GroupType Attribute*

GroupType Attribute	Scope	Bits That Define Type and Scope
GROUP_TYPE_GLOBAL_GROUP	Distribution	0x00000002
GROUP_TYPE_DOMAIN_LOCAL_GROUP	Distribution	0x00000004
GROUP_TYPE_UNIVERSAL_GROUP	Distribution	0x00000008
GROUP_TYPE_SECURITY_ENABLED	Security	0x80000000

8.2 Managing Microsoft Exchange Mailboxes

The Active Directory driver can be configured to create, move and delete Microsoft Exchange mailboxes for users in Active Directory. Mailboxes are managed by setting and removing the value for the homeMDB attribute on the user object. This attribute holds the Distinguished Name of the Exchange Private Message Database (MDB) where the mailbox resides. The driver manages mailboxes on Exchange servers that are in the same domain as the driver only.

There are several different ways to manage Exchange mailboxes. The default configuration manages mailboxes through policy decisions made in the Subscriber Command Transformation policy. When a user meets the given conditions, a mailbox is created, moved or removed. The import file gives you three choices for mailbox management:

- Entitlements
- Policies
- Do not Manage Exchange Mailboxes

When using the entitlement method for provisioning, a user is granted or denied a mailbox based on the entitlement set on the user in the Identity Vault. The entitlement holds the Distinguished Name of the MDB and a state value that tells the driver whether the entitlement is granted or revoked. The entitlement itself is managed by the User Application or the Role-Based Entitlements driver. In either case, the external tool grants (or revokes) the right to the mailbox, the Subscriber Command Transformation policy translates that right into an add-value or remove-value on the homeMDB attribute and the driver shim translates the change to homeMDB into the proper calls to the Exchange management system.

If you are using entitlements and have multiple MDBs in your organization, you use the User Application to decide which MDB is to be assigned to a given user. The Identity Manager Accessory Portlet Reference Guide (http://www.novell.com/documentation/idm) contains the documentation on how to configure multiple MDBs. The role of the Identity Manager driver is to respond to the entitlements placed on the user object, not to put them there. If you are using the User Application, you are given a list of Exchange MDBs to choose from as the workflow item flows through the approval process. If you are using Role-Based entitlements, the MDB is assigned to the group that holds the role for the user.

When using the policy-based method for provisioning, the Subscriber Command Transformation policy uses information about the state of the user object in the Identity Vault to assign the MDB. The driver shim translates the change into the proper calls to the Exchange management system. The default policy uses a simple rule for assigning the mailbox. It assumes that there is only one MDB and that all users that have come this far through the policy chain should be assigned to that MDB. Because the rules for assigning different MDBs vary widely from company to company, the default configuration does not attempt to establish a “right way” of doing it. You implement your own policies simply by changing the default assignment rules. You use DirXML Script if statements to define the conditions for mailbox assignments and the do-set-dest-attribute command for the homeMDB attribute to effect the the change. You can get a list of Exchange MDBs using the `ADManager.exe` tool or by your own means.

When it is not managing Exchange mailboxes, the driver will synchonize the user’s e-mail address and mail nickname.

There are other ways to manage the Exchange mailbox. For instance, you could extend the schema of the Identity Vault to hold the homeMDB information and use basic data sychronization to assign the mailbox to the user in Active Directory. In this case, you would use your own tool to make assignments in the Identity Vault.

The default policy works well for simple mailbox assignment to a single MDB. If you want the policy to reflect more complex rules demanded in your envionment, the policy must be changed.

8.3 Starting, Stopping, or Restarting the Driver

8.3.1 Starting the Driver in Designer

1. Open a project in the Modeler, then right-click the driver line.
2. Select *Live* > *Start Driver*.

8.3.2 Starting the Driver in iManager

1. In iManager, click *Identity Manager* > *Identity Manager Overview*.
2. Browse to the driver set where the driver exists, then click *Search*.
3. Click the upper right corner of the driver icon, then click *Start driver*.

8.3.3 Stopping the Driver in Designer

1. Open a project in the Modeler, then right-click the driver line.
2. Select *Live* > *Stop Driver*.

8.3.4 Stopping the Driver in iManager

1. In iManager, click *Identity Manager* > *Identity Manager Overview*.
2. Browse to the driver set where the driver exists, then click *Search*.
3. Click the upper right corner of the driver icon, then click *Stop driver*.

8.3.5 Restarting the Driver in Designer

1. Open a project in the Modeler, then right-click the driver line.
2. Select *Live* > *Restart Driver*.

8.3.6 Restarting the Driver in iManager

1. In iManager, click *Identity Manager* > *Identity Manager Overview*.
2. Browse to the driver set where the driver exists, then click *Search*.
3. Click the upper right corner of the driver icon, then click *Restart driver*.

8.4 Migrating and Resynchronizing Data

Identity Manager synchronizes data when the data changes. If you want to synchronize all data immediately, you can choose from the following options:

- **Migrate Data from Identity Vault:** Allows you to select containers or objects you want to migrate from the Identity Vault to an application. When you migrate an object, the Identity Manager engine applies all of the Matching, Placement, and Create policies, as well as the Subscriber filter, to the object.
- **Migrate Data into Identity Vault:** Assumes that the remote application can be queried for entries that match the criteria in the publisher filter.
- **Synchronize:** The Identity Manager engine looks in the Subscriber class filter and processes all objects for those classes. Associated objects are merged. Unassociated objects are processed as Add events.

To use one of the options explained above:

1. In iManager, click *Identity Manager* > *Identity Manager Overview*.
2. Browse to and select the driver set where the driver exists, then click *Search*.
3. Click the driver icon, then click the *Migrate* tab.
4. Click the appropriate migration button.

For more information, see Chapter 9, “Synchronizing Objects,” on page 107.

8.5 Using the DirXML Command Line Utility

The DirXML Command Line utility provides command line access to manage the driver. This utility is not a replacement for iManager or Designer. The primary use of this utility is to allow you to create platform-specific scripts to manage the driver.

For example, you could create a shell script on Linux to check the status of the driver. See Appendix D, “DirXML Command Line Utility,” on page 153 for detailed information about the DirXML Command Line utility. For daily tasks, use iManager or Designer.

8.6 Viewing Driver Versioning Information

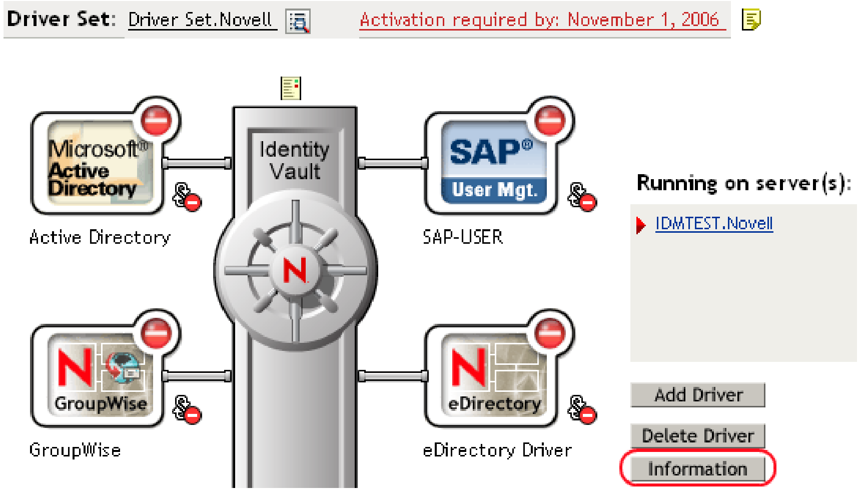
The Versioning Discovery tool only exists in iManager.

- Section 8.6.1, “Viewing a Hierarchical Display of Versioning Information,” on page 93
- Section 8.6.2, “Viewing the Versioning Information as a Text File,” on page 95
- Section 8.6.3, “Saving Versioning Information,” on page 96

8.6.1 Viewing a Hierarchical Display of Versioning Information

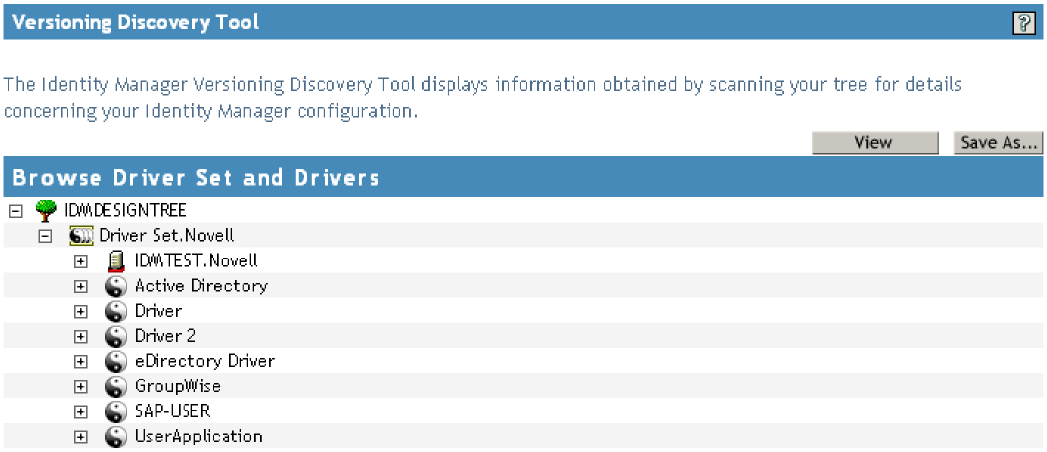
1 To find your Driver Set object in iManager, click *Identity Manager > Identity Manager Overview*, then click *Search*.

2 In the Identity Manager Overview, click *Information.*



You can also select *Identity Manager Utilities > Versions Discovery*, browse to and select the Driver Set object, then click *OK*.

3 View a top-level or unexpanded display of versioning information.



The Identity Manager Versioning Discovery Tool displays information obtained by scanning your tree for details concerning your Identity Manager configuration.

Browse Driver Set and Drivers

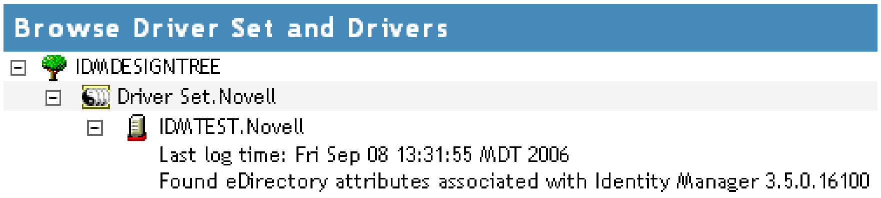
```
⊟ IDMDESIGNTREE
   ⊟ Driver Set.Novell
      ⊞ IDMTEST.Novell
      ⊞ Active Directory
      ⊞ Driver
      ⊞ Driver 2
      ⊞ eDirectory Driver
      ⊞ GroupWise
      ⊞ SAP-USER
      ⊞ UserApplication
```

The unexpanded hierarchical view displays the following:

- The eDirectory™ tree that you are authenticated to
- The Driver Set object that you selected
- Servers that are associated with the Driver Set object

  If the Driver Set object is associated with two or more servers, you can view Identity Manager information on each server.
- Drivers

4 View versioning information related to servers by expanding the server icon.



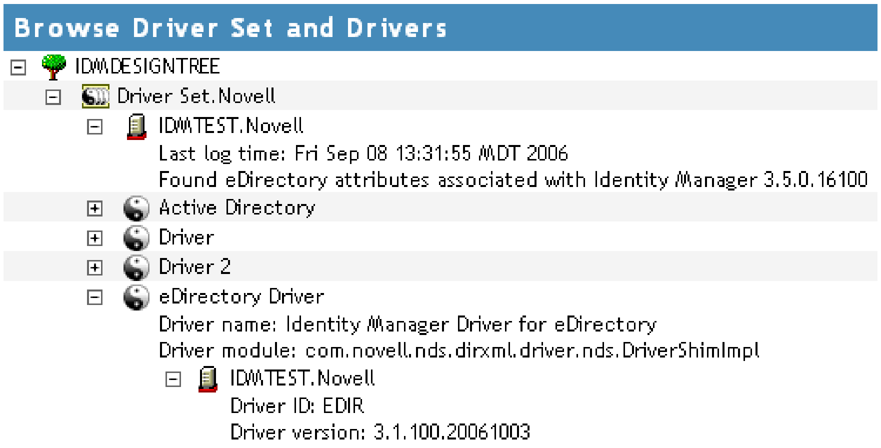
Browse Driver Set and Drivers

```
⊟ IDMDESIGNTREE
    ⊟ Driver Set.Novell
        ⊟ IDMTEST.Novell
            Last log time: Fri Sep 08 13:31:55 MDT 2006
            Found eDirectory attributes associated with Identity Manager 3.5.0.16100
```

The expanded view of a top-level server icon displays the following:

- Last log time
- Version of Identity Manager that is running on the server

5 View versioning information related to drivers by expanding the driver icon.



Browse Driver Set and Drivers

```
⊟ IDMDESIGNTREE
    ⊟ Driver Set.Novell
        ⊟ IDMTEST.Novell
            Last log time: Fri Sep 08 13:31:55 MDT 2006
            Found eDirectory attributes associated with Identity Manager 3.5.0.16100
        ⊞ Active Directory
        ⊞ Driver
        ⊞ Driver 2
        ⊟ eDirectory Driver
            Driver name: Identity Manager Driver for eDirectory
            Driver module: com.novell.nds.dirxml.driver.nds.DriverShimImpl
                ⊟ IDMTEST.Novell
                    Driver ID: EDIR
                    Driver version: 3.1.100.20061003
```

The expanded view of a top-level driver icon displays the following:

- The driver name
- The driver module (for example, com.novell.nds.dirxml.driver.delimitedtext.DelimitedTextDriver)

The expanded view of a server under a driver icon displays the following:

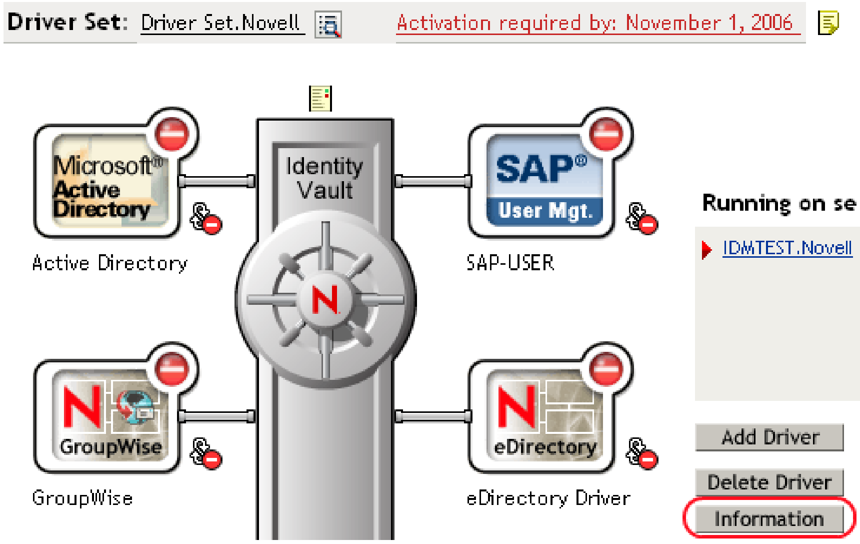
- The driver ID
- The version of the instance of the driver running on that server

8.6.2 Viewing the Versioning Information as a Text File

Identity Manager publishes versioning information to a file. You can view this information in text format. The textual representation is the same information contained in the hierarchical view.

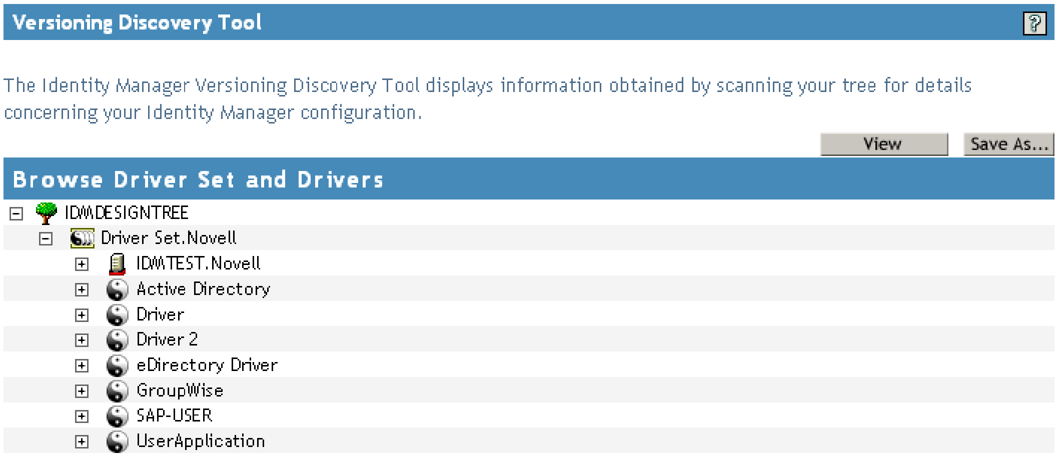
1 To find your Driver Set object in iManager, click *Identity Manager > Identity Manager Overview*, then click *Search.*

2 In the Identity Manager Overview, click *Information*.



You can also select *Identity Manager Utilities > Versioning Discovery*, browse to and select the Driver Set object, then click *Information*.

3 In the Versioning Discovery Tool dialog box, click *View*.



Versioning Discovery Tool

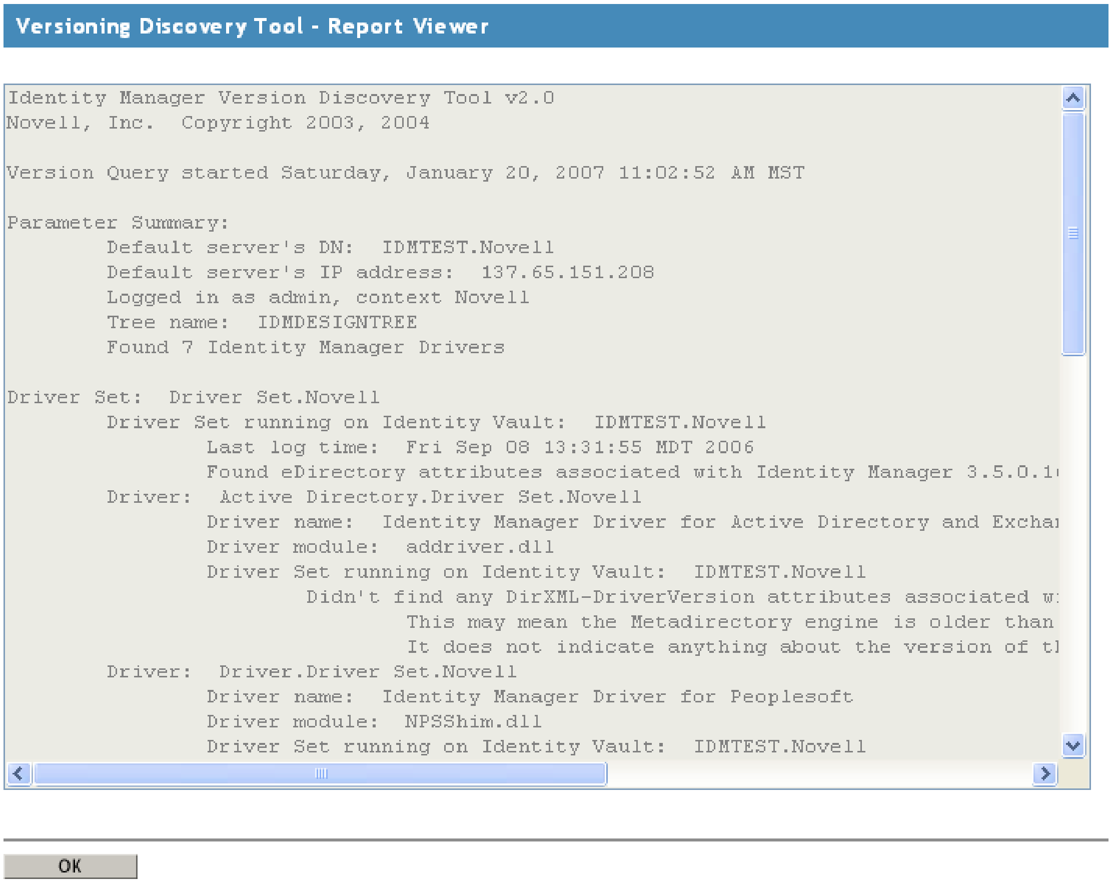
The Identity Manager Versioning Discovery Tool displays information obtained by scanning your tree for details concerning your Identity Manager configuration.

View | Save As...

Browse Driver Set and Drivers

```
⊟ IDMDESIGNTREE
   ⊟ Driver Set.Novell
      ⊞ IDMTEST.Novell
      ⊞ Active Directory
      ⊞ Driver
      ⊞ Driver 2
      ⊞ eDirectory Driver
      ⊞ GroupWise
      ⊞ SAP-USER
      ⊞ UserApplication
```

The information is displayed as a text file in the Report Viewer window.

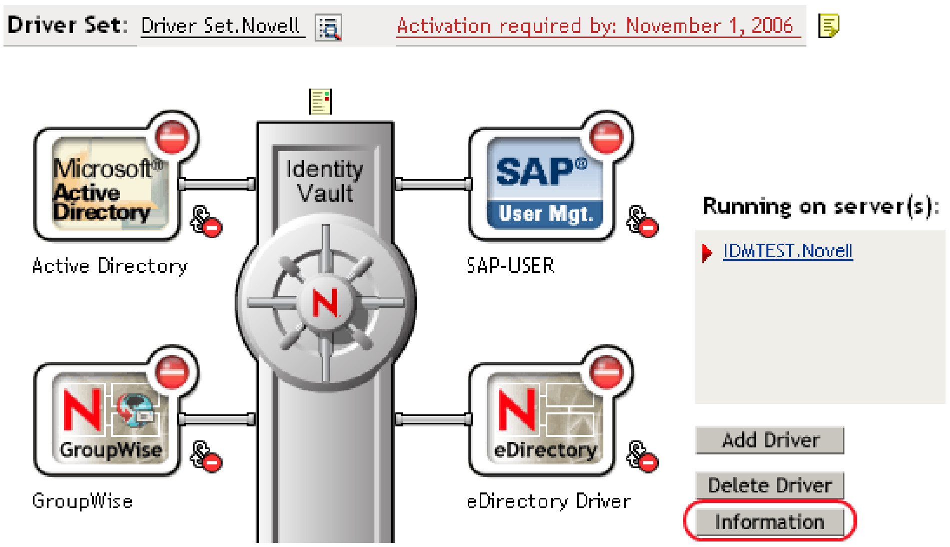


```
Identity Manager Version Discovery Tool v2.0
Novell, Inc.  Copyright 2003, 2004

Version Query started Saturday, January 20, 2007 11:02:52 AM MST

Parameter Summary:
        Default server's DN:  IDMTEST.Novell
        Default server's IP address:  137.65.151.208
        Logged in as admin, context Novell
        Tree name:  IDMDESIGNTREE
        Found 7 Identity Manager Drivers

Driver Set:  Driver Set.Novell
        Driver Set running on Identity Vault:  IDMTEST.Novell
                Last log time:  Fri Sep 08 13:31:55 MDT 2006
                Found eDirectory attributes associated with Identity Manager 3.5.0.1
        Driver:  Active Directory.Driver Set.Novell
                Driver name:  Identity Manager Driver for Active Directory and Excha
                Driver module:  addriver.dll
                Driver Set running on Identity Vault:  IDMTEST.Novell
                        Didn't find any DirXML-DriverVersion attributes associated w
                                This may mean the Metadirectory engine is older than
                                It does not indicate anything about the version of t
        Driver:  Driver.Driver Set.Novell
                Driver name:  Identity Manager Driver for Peoplesoft
                Driver module:  NPSShim.dll
                Driver Set running on Identity Vault:  IDMTEST.Novell
```

8.6.3 Saving Versioning Information

You can save versioning information to a text file on your local or network drive.

1. To find the Driver Set object in iManager, click *Identity Manager* > *Identity Manager Overview*, then click *Search*.
2. In the Identity Manager Overview, click *Information.*



Driver Set: Driver Set.Novell

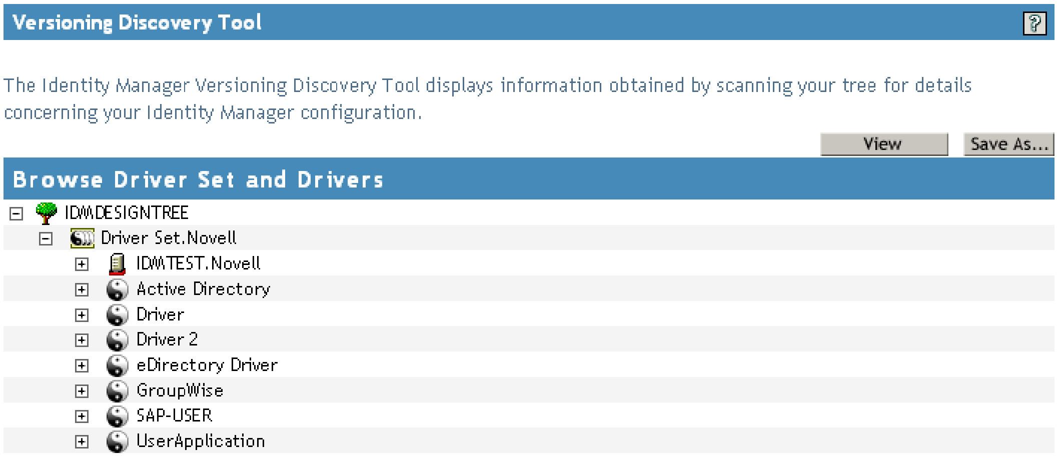
Activation required by: November 1, 2006

Running on server(s):

IDMTEST.Novell

You can also select *Identity Manager Utilities* > *Versioning Discovery*, browse to and select the Driver Set object, then click *Information*.

3 In the Versioning Discovery Tool dialog box, click *Save As*.



Versioning Discovery Tool

The Identity Manager Versioning Discovery Tool displays information obtained by scanning your tree for details concerning your Identity Manager configuration.

Browse Driver Set and Drivers

- IDMDESIGNTREE
  - Driver Set.Novell
    - IDMTEST.Novell
    - Active Directory
    - Driver
    - Driver 2
    - eDirectory Driver
    - GroupWise
    - SAP-USER
    - UserApplication

4 In the File Download dialog box, click *Save*.

5 Navigate to the desired directory, type a filename, then click *Save*.

Identity Manager saves the data to a text file.

8.7 Reassociating a Driver Set Object with a Server Object

The driver set object should always be associated with a server object. If the driver set is not associated with a server object, none of the drivers in the driver set can start.

If the link between the driver set object and the server object becomes invalid, you see one of the following conditions:

- When upgrading eDirectory your Identity Manager server, you get the error UniqueSPIException error -783.
- No server is listed next to the driver set in the Identity Manager Overview window.
- A server is listed next to the driver set in the Identity Manager Overview window, but the name is garbled text.

To resolve this issue, disassociate the driver set object and the server object, then reassociate them.

1. In iManager click *Identity Manager > Identity Manager Overview*, then click *Search* to find the driver set object that the driver should be associated with.
2. Click the *Remove server* icon, then click *OK*.
3. Click the *Add server* icon, then browse to and select the server object.
4. Click *OK*.

8.8 Changing the Driver Configuration

If you need to change the driver configuration, Identity Manager allows you to make the change through iManager or Designer.

To change the driver configuration in iManager:

1. Click *Identity Manager > Identity Manager Overview*, then click *Search* to search for the driver set that is associated with the driver.
2. Browse to the driver, then click the upper right corner of the driver icon.
3. Click *Edit Properties*.

To change the driver configuration in Designer:

1. Open a project in the Modeler, then right-click the driver line and select *Properties.*

For a listing of all of the configuration fields, see Appendix E, "Properties of the Driver," on page 167.

8.9 Storing Driver Passwords Securely with Named Passwords

Identity Manager allows you to store multiple passwords securely for a particular driver. This functionality is referred to as Named Passwords. Each different password is accessed by a key, or name.

You can also use the Named Passwords feature to store other pieces of information securely, such as a user name.

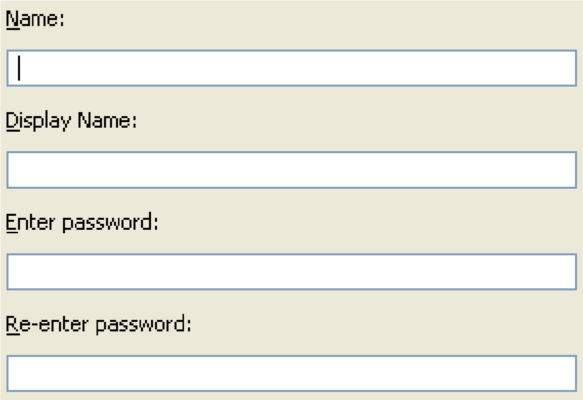
To use a Named Password in a driver policy, you refer to it by the name of the password, instead of using the actual password, and the Metadirectory engine sends the password to the driver. The method described in this section for storing and retrieving Named Passwords can be used with any driver without making changes to the driver shim.

- Section 8.9.1, "Using Designer to Configure Named Passwords," on page 99
- Section 8.9.2, "Using iManager to Configure Named Passwords," on page 99
- Section 8.9.3, "Using Named Passwords in Driver Policies," on page 101
- Section 8.9.4, "Using the DirXML Command Line Utility to Configure Named Passwords," on page 101

8.9.1 Using Designer to Configure Named Passwords

1 Right-click the driver object, then select *Properties*.

2 Select *Named Password*, then click *New*.



Name:

Display Name:

Enter password:

Re-enter password:

3 Specify the *Name* of the Named Password.

4 Specify the *Display name* of the Named Password.

5 Specify the Named Password, then re-enter the password.

6 Click *OK* twice.

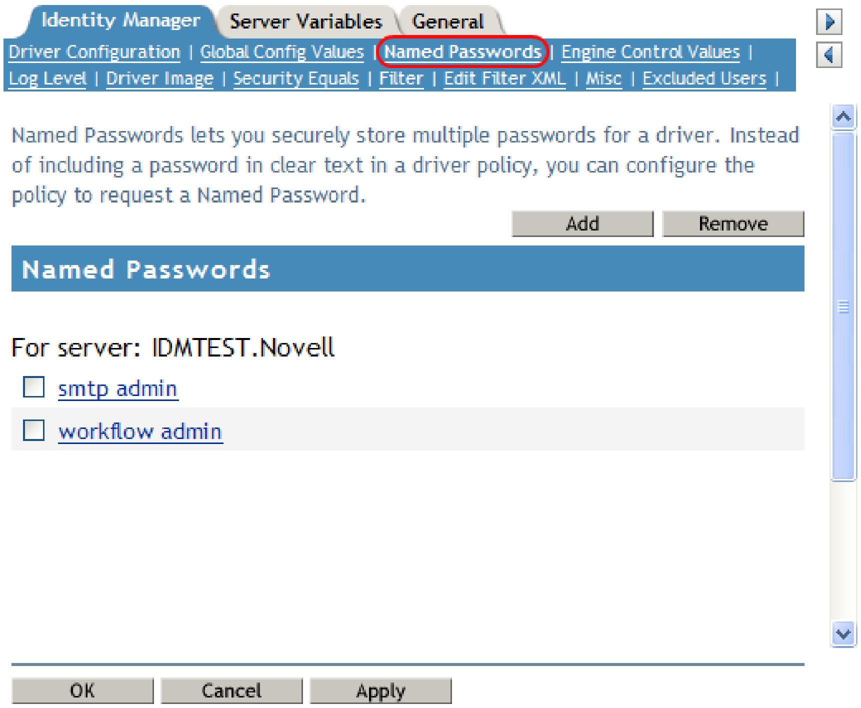
8.9.2 Using iManager to Configure Named Passwords

1 Click *Identity Manager > Identity Manager Overview*, then click *Search* to search for the driver set that is associated with the driver.

2 In the Identity Manager Overview, click the upper right corner of the driver icon, then click *Edit properties*.

3 On the Modify Object page on the Identity Manager tab, click *Named Passwords*.

The Named Passwords page appears, listing the current Named Passwords for this driver. If you have not set up any Named Passwords, the list is empty.


Identity Manager
Server Variables
General
Driver Configuration | Global Config Values | Named Passwords | Engine Control Values |
Log Level | Driver Image | Security Equals | Filter | Edit Filter XML | Misc | Excluded Users |
Add
Remove
OK
Cancel
Apply

Named Passwords lets you securely store multiple passwords for a driver. Instead of including a password in clear text in a driver policy, you can configure the policy to request a Named Password.

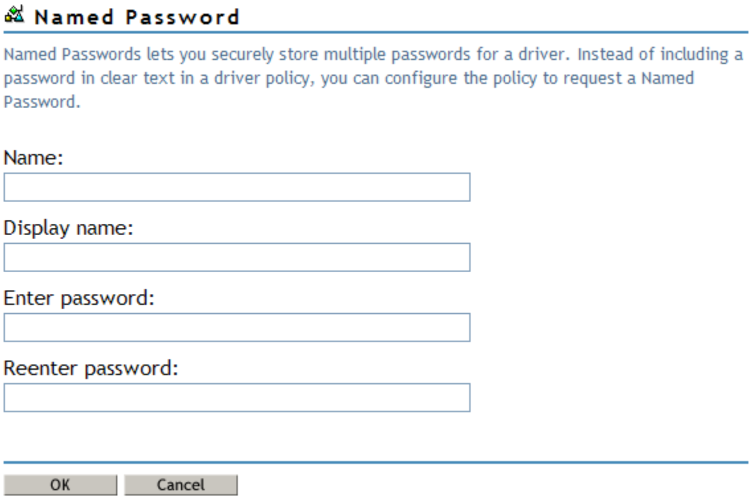
Named Passwords

For server: IDMTEST.Novell

□ smtp admin

□ workflow admin

4 To add a Named Password, click *Add*, complete the fields, then click *OK*.



Named Passwords lets you securely store multiple passwords for a driver. Instead of including a password in clear text in a driver policy, you can configure the policy to request a Named Password.

Name:

Display name:

Enter password:

Reenter password:

5 Specify a name, display name and a password, then click *OK* twice.

You can use this feature to store other kinds of information securely, such as a username.

6 Click *OK* to restart the driver and have the changes take effect.

7 To remove a Named Password, select the password name, then click *Remove*.

The password is removed without prompting you to confirm the action.

8.9.3 Using Named Passwords in Driver Policies

- "Using the Policy Builder" on page 101
- "Using XSLT" on page 101

Using the Policy Builder

Policy Builder allows you to make a call to a Named Password. Create a new rule and select Named Password as the condition, then set an action depending upon if the Named Password is available or not available.

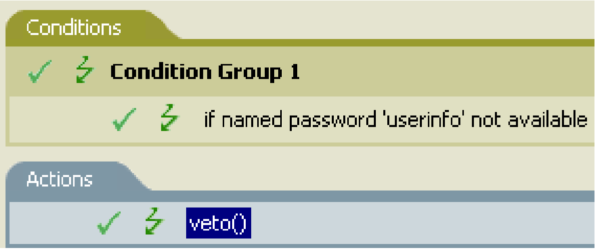
1. In Designer, launch Policy Builder, right-click, then click *New* > *Rule*.
2. Specify the name of the rule, then click *Next*.
3. Select the condition structure, then click *Next*.
4. Select *named password* for the *Condition*.
5. Browse to and select the Named Password that is stored on the driver.

   In this example, it is *userinfo*.
6. Select whether the Operator is available or not available.
7. Select an action for the *Do* field.

   In this example, the action is *veto*.

The example indicates that if the userinfo Named Password is not available, then the event is vetoed.

Figure 8-1 *A Policy Using Named Passwords*



Conditions

✓ Condition Group 1

✓ if named password 'userinfo' not available

Actions

✓ veto()

Using XSLT

The following example shows how a Named Password can be referenced in a driver policy on the Subscriber channel in XSLT:

```
<xsl:value-of
select="query:getNamedPassword($srcQueryProcessor,'mynamedpassword')"
xmlns:query="http://www.novell.com/java/
com.novell.nds.dirxml.driver.XdsQueryProcessor/>
```

8.9.4 Using the DirXML Command Line Utility to Configure Named Passwords

- "Creating a Named Password in the DirXML Command Line Utility" on page 102
- "Using the DirXML Command Line Utility to Remove a Named Password" on page 103

Creating a Named Password in the DirXML Command Line Utility

1 Run the DirXML Command Line utility.

For information, see Appendix D, "DirXML Command Line Utility," on page 153.

2 Enter your username and password.

The following list of options appears.

```
DirXML commands

 1: Start driver
 2: Stop driver
 3: Driver operations...
 4: Driver set operations...
 5: Log events operations...
 6: Get DirXML version

 7: Job operations...
99: Quit

Enter choice:
```

3 Enter `3` for driver operations.

A numbered list of drivers appears.

4 Enter the number for the driver you want to add a Named Password to.

The following list of options appears.

```
Select a driver operation for:
driver_name

 1: Start driver
 2: Stop driver
 3: Get driver state
 4: Get driver start option
 5: Set driver start option
 6: Resync driver
 7: Migrate from application into DirXML
 8: Submit XDS command document to driver

 9: Submit XDS event document to driver

10: Queue event for driver
11: Check object password
12: Initialize new driver object
13: Passwords operations
14: Cache operations
99: Exit

Enter choice:
```

5 Enter `13` for password operations.

The following list of options appears.

```
Select a password operation

 1: Set shim password
 2: Reset shim password

 3: Set Remote Loader password

 4: Clear Remote Loader password
 5: Set named password
 6: Clear named password(s)
 7: List named passwords
 8: Get passwords state
99: Exit

Enter choice:
```

6. Enter `5` to set a new Named Password.

   The following prompt appears:

   ```
   Enter password name:
   ```

7. Enter the name by which you want to refer to the Named Password.
8. Enter the actual password that you want to secure at the following prompt:

   ```
   Enter password:
   ```

   The characters you type for the password are not displayed.

9. Confirm the password by entering it again at the following prompt:

   ```
   Confirm password:
   ```

10. After you enter and confirm the password, you are returned to the password operations menu.
11. After completing this procedure, you can use the 99 option twice to exit the menu and quit the DirXML Command Line Utility.

Using the DirXML Command Line Utility to Remove a Named Password

This option is useful if you no longer need Named Passwords that you previously created.

1. Run the DirXML Command Line utility.

   For information, see Appendix D, "DirXML Command Line Utility," on page 153.

2. Enter your username and password.

   The following list of options appears.

   ```
   DirXML commands

    1: Start driver
    2: Stop driver
    3: Driver operations...
    4: Driver set operations...
    5: Log events operations...
    6: Get DirXML version

    7: Job operations
   99: Quit

   Enter choice:
   ```

3. Enter `3` for driver operations.

   A numbered list of drivers appears.

4. Enter the number for the driver you want to remove Named Passwords from.

   The following list of options appears.

   ```
   Select a driver operation for:
   driver_name
   ```

```
 1: Start driver
 2: Stop driver
 3: Get driver state
 4: Get driver start option
 5: Set driver start option
 6: Resync driver
 7: Migrate from application into DirXML
 8: Submit XDS command document to driver

 9: Submit XDS event document to driver

10: Queue event for driver
11: Check object password
12: Initialize new driver object
13: Passwords operations
14: Cache operations
99: Exit

Enter choice:
```

5 Enter `13` for password operations.

The following list of options appears.

```
Select a password operation

 1: Set shim password
 2: Reset shim password

 3: Set Remote Loader password

 4: Clear Remote Loader passwor
 5: Set named password
 6: Clear named password(s)
 7: List named passwords

 8: Get passwords state
99: Exit

Enter choice:
```

6 (Optional) Enter `7` to see the list of existing Named Passwords.

The list of existing Named Passwords is displayed.

This step can help you make sure you are removing the correct password.

7 Enter `6` to remove one or more Named Passwords.

8 Enter `No` to remove a single Named Password at the following prompt:

```
Do you want to clear all named passwords? (yes/no):
```

9 Enter the name of the Named Password you want to remove at the following prompt:

```
Enter password name:
```

After you enter the name of the Named Password you want to remove, you are returned to the password operations menu:

```
Select a password operation

 1: Set shim password
 2: Reset shim password

 3: Set Remote Loader password

 4: Clear Remote Loader password
 5: Set named password
 6: Clear named password(s)
 7: List named passwords
 8: Get passwords state
99: Exit
```

```
Enter choice:
```

10 (Optional) Enter `7` to see the list of existing Named Passwords.

This step lets you verify that you have removed the correct password.

11 After completing this procedure, you can use the 99 option twice to exit the menu and quit the DirXML Command Line utility.

8.10 Adding a Driver Heartbeat

The driver heartbeat is a feature of the Identity Manager drivers that ship with Identity Manager 2 and later. Its use is optional. The driver heartbeat is configured by using a driver parameter with a time interval specified. If a heartbeat parameter exists and has an interval value other than 0, the driver sends a heartbeat document to the Metadirectory engine if there is no communication on the Publisher channel for the specified interval of time.

The intent of the driver heartbeat is to give you a trigger to allow you to initiate an action at regular intervals, if the driver does not communicate on the Publisher channel as often as you want the action to occur. To take advantage of the heartbeat, you must customize your driver configuration or other tools. The Metadirectory engine accepts the heartbeat document but does not take any action because of it.

For most drivers, a driver parameter for heartbeat is not used in the sample configurations, but you can add it.

A custom driver that is not provided with Identity Manager can also provide a heartbeat document, if the driver developer has written the driver to support it.

To configure the heartbeat:

1 In iManager, click *Identity Manager* > *Identity Manager Overview*.

2 Browse to and select your driver set object, then click *Search*.

3 In the Identity Manager Overview, click the upper right corner of the driver icon, then click *Edit properties*.

4 On the *Identity Manager* tab, click *Driver Configuration*, scroll to *Driver Parameters*, then look for Heart Beat or a similar display name.

If a driver parameter already exists for heartbeat, you can change the interval and save the changes, and configuration is then complete.

The value of the interval cannot be less than 1. A value of 0 means the feature is turned off.

The unit of time is usually minutes; however, some drivers might choose to implement it differently, such as using seconds.

5 If a driver parameter does not exist for heartbeat, click *Edit XML*.

6 Add a driver parameter entry like the following example, as a child of `<publisher-options>`. (For an AD driver, make it a child of `<driver-options>`.)

```
<pub-heartbeat-interval display-name="Heart Beat">10</pub-heartbeat-
interval>
```

If the driver does not produce a heartbeat document after being restarted, check the placement of the driver parameter in the XML.

7 Save the changes, and make sure the driver is stopped and restarted.

After you have added the driver parameter, you can edit the time interval by using the graphical view. Another option is to create a reference to a global configuration value (GCV) for the time interval. Like other global configuration values, the driver heartbeat can be set at the driver set level instead of on each individual driver object. If a driver does not have a particular global configuration value, and the driver set object does have it, the driver inherits the value from the driver set object.

Synchronizing Objects

9

This section explains driver and object synchronization in DirXML® 1.1a, Identity Manager 2.0, and Identity Manager 3.x. Driver synchronization was not available for DirXML 1.0 and DirXML 1.1.

After the driver is created, instead of waiting for objects to be modified or created, the data between the two connected systems can be sent through the synchronization process.

9.1 What Is Synchronization?

The actions commonly referred to as "synchronization" in Identity Manager refer to several different but related actions:

- Synchronization (or merging) of attribute values of an object in the Identity Vault with the corresponding attribute values of an associated object in a connected system.
- Migration of all Identity Vault objects and classes that are included in the filter on the Subscriber channel.
- Generation of the list of objects to submit to the driver's Subscriber channel for synchronization or migration in response to a user request (a manual synchronization).
- Generation of the list of objects to submit to the driver's Subscriber channel for synchronization or migration in response to enabling a formerly disabled driver, or in response to a cache error.

9.2 When Is Synchronization Done?

The Metadirectory engine performs object synchronization or merging in the following circumstances:

- A `<sync>` event element is submitted on the Subscriber or Publisher channel.
- A `<sync>` event element is submitted on the Subscriber channel in the following circumstances:
  - The state of the object's association value is set to "manual" or "migrate." (This causes an eDirectory™ event, which in turn causes the Identity Manager caching system to queue an object synchronization command in the affected driver's cache.)
  - An object synchronization command is read from the driver's cache.

- A <sync> event element is submitted on the Publisher channel in the following circumstances:
  - A driver submits a <sync> event element. No known driver currently does this.
  - The Metadirectory engine submits a <sync> event element for each object found as the result of a migrate-into-NDS query. These <sync> events are submitted using the Subscriber thread, but are processed using the Publisher channel filter and policies.
- An <add> event (real or synthetic) is submitted on a channel and the channel Matching policy finds a matching object in the target system.
- An <add> event with an association is submitted on the Subscriber channel. This normally occurs only in exceptional cases, such as the bulk load of objects into eDirectory with DirXML-Associations attribute values.
- An <add> event is submitted on the Publisher channel and an object is found in eDirectory that already has the association value reported with the <add> event.

The Metadirectory engine generates synchronization requests for zero or more objects in the following cases:

- The user issues a manual driver synchronization request. This corresponds to the *Resync* button in the Driver Set property page in ConsoleOne®, or to the *Synchronize* button on the iManager Identity Manager Driver Overview page.
- The Metadirectory engine encounters an error with the driver's cache and cannot recover from the cache error. The driver's cache is deleted and the engine generates object synchronization commands as detailed in Section 9.3, "How Does the Metadirectory Engine Decide Which Object to Synchronize?," on page 108.

9.3 How Does the Metadirectory Engine Decide Which Object to Synchronize?

The Metadirectory engine processes both manually initiated and automatically initiated synchronization requests in the same manner. The only difference in the processing of manually initiated versus automatically initiated driver synchronization requests is the starting filter time used to filter objects being considered for synchronization.

The starting filter time is used to filter objects that have modification or creation times that are older than the starting time specified in the synchronization request.

For automatically initiated driver synchronization, the starting filter time is obtained from the time stamps of cached eDirectory events. In particular, the starting filter time is the earliest time for the cached events that haven't yet been successfully processed by the driver's Subscriber channel.

For manually initiated driver synchronization, the default starting filter time is the earliest time in the eDirectory database. In Identity Manager 2 and Identity Manager 3, an explicit starting filter time can also be set. In DirXML 1.1a there is no facility to set the starting filter time value for synchronization when manually initiating driver synchronization.

The Metadirectory engine creates a list of objects to be synchronized on the Subscriber channel in the following manner:

1. It finds all objects that:
   - Have an entry modification time stamp greater than or equal to the starting filter time

and

- Exist in the filter on the Subscriber channel.

2. It finds all objects that have an entry creation time stamp greater than or equal to the starting filter time.
3. It adds a `synchronize object` command to the driver cache for each unique object found that has an entry modification time stamp greater than or equal to the starting filter time and all objects and classes that are in the Subscriber filter channel in the driver being synchronized.

9.4 How Does Synchronization Work?

After the Metadirectory engine determines that an object is to be synchronized, the following processes occur:

1. Each system (the Identity Vault and the connected system) is queried for all attribute values in the appropriate filters.
   - eDirectory is queried for all values in the Subscriber filter, and for values that are marked for synchronization in Identity Manager 2.x and Identity Manager 3.x.
   - The connected system is queried for all values in the Publisher filter, and for values that are marked for synchronization in Identity Manager 2.x and Identity Manager 3.x.
2. The returned attribute values are compared and modification lists are prepared for the Identity Vault and the connected system according to Table 9-1 on page 110, Table 9-2 on page 111, and Table 9-3 on page 113.

   In the tables the following pseudo-equations are used:

   - "Left = Right" indicates that the left side receives all values from the right side.
   - "Left = Right[1]" indicates that the left side receives one value from the right side. If there is more than one value, it is indeterminate.
   - "Left += Right" indicates that the left side adds the right side values to the left side's existing values.
   - "Left = Left + Right" indicates that the left sides receives the union of the values of the left and right sides.

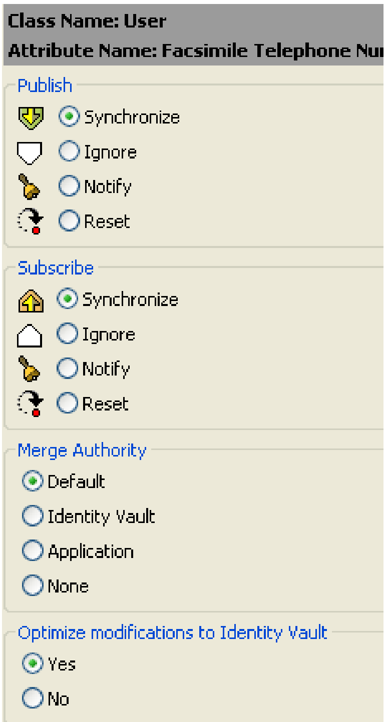
There are three different combinations of selected items in the filter, and each one creates a different output.

- Section 9.4.1, "Scenario One," on page 109
- Section 9.4.2, "Scenario Two," on page 111
- Section 9.4.3, "Scenario Three," on page 112

9.4.1 Scenario One

The attribute is set to *Synchronize* on the Publisher and Subscriber channels, and the merge authority is set to *Default*.

Figure 9-1 *Scenario One*



Class Name: User
Attribute Name: Facsimile Telephone Nu

Publish
● Synchronize
○ Ignore
○ Notify
○ Reset

Subscribe
● Synchronize
○ Ignore
○ Notify
○ Reset

Merge Authority
● Default
○ Identity Vault
○ Application
○ None

Optimize modifications to Identity Vault
● Yes
○ No

The following table contains the values that the Metadirectory engine synchronizes when the attribute is sent through a filter that is set to the configuration for Scenario One. The table shows different outputs depending upon whether the attribute comes from the Identity Vault or the Application, if the attribute is single-valued or multi-valued, and if the attribute is empty or non-empty.

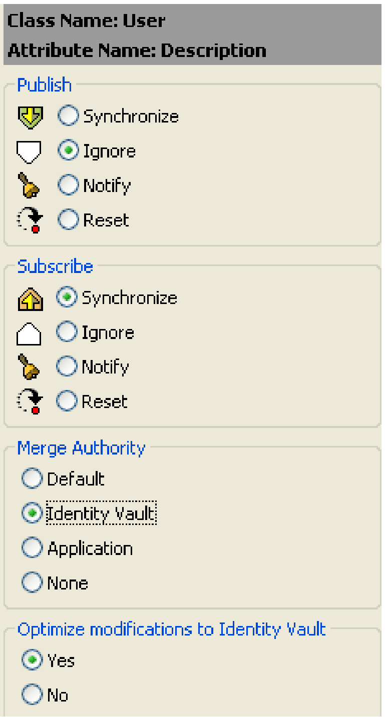
Table 9-1 *Output of Scenario One*

	Identity Vault single-valued empty	Identity Vault single-valued non-empty	Identity Vault multi-valued empty	Identity Vault multi-valued non-empty
Application single-valued empty	No change	App = Identity Vault	No change	App = Identity Vault[1]
Application single-valued non-empty	Identity Vault = App	App = Identity Vault	Identity Vault = App	Identity Vault + = App
Application multi-valued empty	No change	App = Identity Vault	No change	App = Identity Vault
Application multi-valued non-empty	Identity Vault = App[1]	App + = Identity Vault	Identity Vault = App	App = App + Identity Vault Identity Vault = App + Identity Vault

9.4.2 Scenario Two

The attribute is set to *Synchronize* only on the Subscriber channel, or it is set to *Synchronize* on both the Subscriber and Publisher channels. The merge authority is set to *Identity Vault*.

Figure 9-2 *Scenario Two*



Class Name: User
Attribute Name: Description

Publish
- ○ Synchronize
- ● Ignore
- ○ Notify
- ○ Reset

Subscribe
- ● Synchronize
- ○ Ignore
- ○ Notify
- ○ Reset

Merge Authority
- ○ Default
- ● Identity Vault
- ○ Application
- ○ None

Optimize modifications to Identity Vault
- ● Yes
- ○ No

The following table contains the values that the Metadirectory engine synchronizes when the attribute is sent through a filter that is set to the configuration for Scenario Two. The table shows different outputs depending upon whether the attribute comes from the Identity Vault or the Application, if the attribute is single-valued or multi-valued, and if the attribute is empty or non-empty.

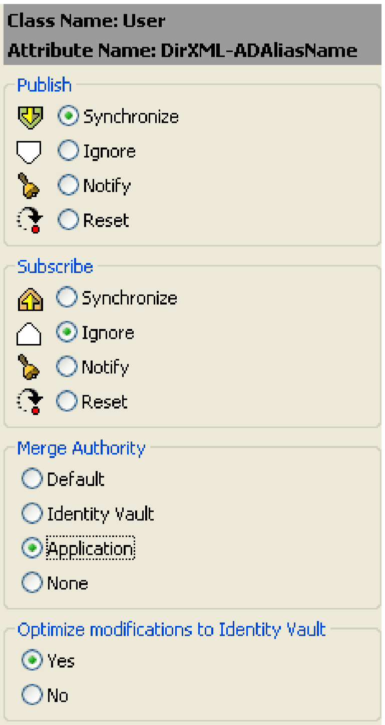
Table 9-2 *Output of Scenario Two*

	Identity Vault single-valued empty	**Identity Vault single-valued non-empty**	**Identity Vault multi-valued empty**	**Identity Vault multi-valued non-empty**
Application single-valued empty	No change	App = Identity Vault	No change	App = Identity Vault[1]
Application single-valued empty	App = empty	App = Identity Vault	Identity Vault = App	App = Identity Vault[1]
Application multi-valued empty	No change	App = Identity Vault	No change	App = Identity Vault

	Identity Vault single-valued empty	Identity Vault single-valued non-empty	Identity Vault multi-valued empty	Identity Vault multi-valued non-empty
Application multi-valued non-empty	App = empty	App = Identity Vault	App = empty	App = Identity Vault

9.4.3 Scenario Three

The attribute is set to *Synchronize* on the Publisher channel or the merge authority is set to *Application*.



Class Name: User
Attribute Name: DirXML-ADAliasName

Publish
● Synchronize
○ Ignore
○ Notify
○ Reset

Subscribe
○ Synchronize
● Ignore
○ Notify
○ Reset

Merge Authority
○ Default
○ Identity Vault
● Application
○ None

Optimize modifications to Identity Vault
● Yes
○ No

Figure 9-3 *Scenario Three*

The following table contains the values that the Metadirectory engine synchronizes when the attribute is sent through a filter that is set to the configuration for Scenario Three. The table shows different outputs depending upon whether the attribute comes from the Identity Vault or the Application, if the attribute is single-valued or multi-valued, and if the attribute is empty or non-empty.

***Table 9-3** Output of Scenario Three*

	Identity Vault single-valued empty	**Identity Vault single-valued non-empty**	**Identity Vault multi-valued empty**	**Identity Vault multi-valued non-empty**
Application single-valued empty	No change	Identity Vault = empty	No change	Identity Vault = empty
Application single-valued non-empty	Identity Vault = App	Identity Vault = App	Identity Vault = App	Identity Vault = App
Application multi-valued empty	No change	Identity Vault = empty	No change	Identity Vault = empty
Application multi-valued non-empty	Identity Vault = App[1]	Identity Vault = App[1]	Identity Vault = App	Identity Vault = App

Troubleshooting 10

10.1 Changes Are Not Synchronizing from the Publisher or Subscriber

To synchronize changes in Active Directory, the account used by the Identity Manager driver must have the proper rights set up. For information on the necessary rights, see Section 2.4, “Creating an Administrative Account,” on page 28.

If you use the default policies, you must also meet the requirements for the Create, Match, and Placement policies. For information on default policy requirements, see “Policies” on page 15.

The attribute dirxml-uACLockout is not synchronized on the Publisher channel.

10.2 Using Characters Outside the Valid NT Logon Names

The default Subscriber creation policy generates an NT Logon Name (also known as the sAMAccountName and the Pre-Windows 2000 Logon Name) based on the relative distinguished name (RDN) of the account in the Identity Vault. The NT Logon name uses a subset of the ASCII character set. The default policy strips any character outside of the valid range before creating an object in Active Directory.

If the policy doesn’t satisfy the business rules of your company, you can change the policy after import. Businesses that use Identity Vault account names outside of the traditional ASCII character set should pay particular attention to this policy.

10.3 Synchronizing c, co, and countryCode Attributes

When you use the Active Directory management console to select a country for a user, three attributes are set:

Table 10-1 *Attributes for Country*

Attribute	Description
c	Contains a two-character country code as defined by the ISO.
co	Contains a longer name for the country.
countryCode	Contains a numeric value (also defined by the ISO) that represents the country.

Because the ISO-defined numeric country codes are intended for use by applications that can't handle alphabetic characters, the default schema in the Identity Vault includes c and co but not countryCode.

Identity Manager is capable of mapping c and co. It can also map countryCode if you add a similar attribute to the eDirectory schema.

Active Directory's management console tries to keep all three of these attributes synchronized, so that when you set the country in the console, all three attributes have appropriate values. Some administrators might want a similar behavior when the attribute is set through Identity Manager. For example, you might want to configure the driver so that even though only c is in the Filter, co and countryCode are also set when a change for c is sent on the Subscriber channel.

10.4 Synchronizing Operational Attributes

Operation attributes are attributes that are maintained by an LDAP server that contains special operational information. Operation attributes are read-only. They can't be synchronized or changed.

10.5 Password Complexity on Windows 2003

Passwords must meet criteria that the password policies specify.

Complexities and requirements in Windows 2000/2003 password policies are different from complexities and requirements in eDirectory.

If you plan to use Password Synchronization, create and use passwords that match the rules of complexity in both Active Directory and eDirectory™. Otherwise, the passwords fail.

TIP: Make the password policies for both systems as similar to each other as you can. In a lab environment, disable strong-password functionality on Windows 2003 servers before installing the Active Directory driver. After the Active Directory driver is working properly, make sure that passwords used in eDirectory and Active Directory satisfy the rules of complexity for both systems. Then re-enable strong-password functionality on the Windows 2003 server.

For troubleshooting tips, see TID 10083320 (http://support.novell.com/cgi-bin/search/searchtid.cgi?/10083320.htm).

10.6 Tips on Password Synchronization

We recommend that you use a secure connection when you are synchronizing passwords. Vulnerable connections are between the following:

- The Metadirectory engine and the Remote Loader
- The Remote Loader and Active Directory

  This is true only when you run the Remote Loader remotely from the domain controller that you're connecting to.
- The Metadirectory engine and Active Directory when you aren't using the Remote Loader

  This is true only if the domain controller isn't local to this machine.

You can create a secure connection by doing one or more of the following:

- Configure SSL between the Metadirectory engine and the Remote Loader
- Run the Remote Loader on the domain controller
- Configure SSL between the driver shim and Active Directory

  This doesn't apply if you are running the driver on the domain controller that you're connecting to.

For password synchronization to work when the driver shim isn't running on the domain controller, you must have SSL configured.

10.6.1 Providing Initial Passwords

If you see an error about a password not complying when a user is initially created, you need to check your password policies.

For example, perhaps you want the Active Directory driver to provide the initial password for a user when the Active Directory driver creates a User object in the Identity Vault. When a user is created, the driver shim creates the user and then sets the password.

Because adding the user and setting the password are done separately, the new user in this example receives the default password, even if only momentarily. The password is soon updated because the Active Directory driver sends it immediately after adding the user.

If the default password doesn't comply with the eDirectory Password Policy for the user, an error is displayed. For example, if a default password that was created by using the user's surname is too short to comply with the Password Policy, you might see a -216 error saying that the password is too short. However, the situation is soon rectified if the Active Directory driver then sends an initial password that does comply.

Regardless of the driver you are using, if you want a connected system that is creating User objects to provide the initial password, consider doing one of the following:

- Change the policy on the Publisher channel that creates default passwords, so that default passwords conform to the Password policies (created by using the *Manage Password Policies* option in Password Management) that have been defined for your organization in the Identity Vault. When the initial password comes from the authoritative application, it replaces the default password.

  This option is preferable. We recommend that a default password policy exist in order to maintain a high level of security within the system.
- Remove the policy on the Publisher channel that creates default password. In the sample configuration, this policy is provided in the Command Transformation policy set. Adding a user without a password is allowed in eDirectory. The assumption for this option is that the password for the newly created User object eventually comes through the Publisher channel, so the user object exists without a password only for a short time.

These measures are especially important if the initial password doesn't come with the Add event, but comes in a subsequent event.

10.7 Where to Set the SSL Parameter

The SSL parameter in the driver configuration is for SSL between the Active Directory driver and Active Directory. It isn't for SSL between the Metadirectory engine and the Remote Loader. See "Encryption" on page 24.

10.8 The Active Directory Account Is Disabled after a User Add on the Subscriber Channel

The default configuration maps the Identity Vault Logon Disabled attribute to the dirxml-uACAccountDisable bit of the userAccountControl attribute in Active Directory. A Subscriber Add operation might set Logon Disabled to False (account enabled), but the Publisher loopback of the Add operation reports that Logon Disabled is True (account disabled).

Additionally, inspecting the object in Active Directory might show that the account is disabled. This happens in part because of the way that the driver creates objects in Active Directory and in part because of a mismatch of policies between the driver and Active Directory itself.

10.8.1 Account Disabled in Active Directory Users and Computers

If the account remains disabled in Active Directory after the provisioning cycle completes, you might have a mismatch between policies configured for the driver and policies enforced by Active Directory.

For example, consider a Password Required policy. If a user Add operation contains an invalid password (or no password at all), the account created in Active Directory should be disabled. But Active Directory might set the dirxml-uACPasswordNotRequired bit in userAccountControl without the driver's knowledge.

This causes the logon enable action of the Add operation to fail if the Add operation does not include a policy for dirxml-uACPasswordNotRequired. Therefore, the account stays disabled.

Later (perhaps almost immediately because of a Merge operation), the driver might attempt to enable the account again by setting Logon Disabled to False. If you want to override the Active Directory policy and ensure that accounts always require a password, you should set dirxml-uACPasswordNotRequired to False whenever Logon Disabled changes on the Subscriber channel.

10.9 Moving a Parent Mailbox to a Child Domain

If you move a parent mailbox to a mailbox store in a child domain by changing a user's homeMDB attribute, the driver fails the move. The error code returned is 0x80072030.

This error occurs on inter-domain moves. Moving an Exchange parent mailbox to a child domain isn't supported.

10.10 Restoring Active Directory

When you need to restore some or all of Active Directory, the driver might pick up interim events and perform unwanted actions on the Identity Vault. To restore safely, temporarily disable the driver during the restore operation and then bring the Identity Vault back into synchronization with Active Directory.

1. Disable the driver.
2. Delete the Dirxml-DriverStorage attribute on the driver object in the Identity Vault.
3. Restore Active Directory.
4. Set the Active Directory driver to Manual or Automatic startup.
5. Start the driver.
6. Re-migrate to find unassociated objects.

10.11 Moving the Driver to a Different Domain Controller

You can configure the driver to synchronize against a different domain controller by changing the driver Authentication Context parameter. When you restart the driver, the state information that the driver uses to track changes in Active Directory is invalid, and Active Directory might replay a large number of old events to bring the state back to the current time.

You can avoid this replay by removing the driver state information while updating the Authentication Context:

1. Stop the driver.
2. Delete the Dirxml-DriverStorage attribute on the Driver object in the Identity Vault.
3. Update the Authentication Context parameter.
4. Start the driver.

   This causes a resynchronization of associated objects in the Identity Vault.
5. Re-migrate to find unassociated objects in Active Directory.

10.12 Migrating from Active Directory

When migrating from Active Directory to the Identity Vault, you need to be concerned about object contianment, DN references, and search limits on the Active Directory server. The general strategy for dealing with containment is to migrate containers first, objects that might be members of groups (including user objects) second, and groups last. If you have a moderately large number of objects to migrate, you need to adjust your strategy to handle the LDAP search constraints configured on the Active Directory server. You can change the constraints on the LDAP server or adjust your migration to get only a subset of objects each time (for instance, migrating container by container or migrating objects starting with A, B, and etc.).

10.13 Setting LDAP Server Search Constraints

Following is a terminal session showing you how to use `ntdsutil.exe` to change the LDAP search parameters on your domain controller. You need only change these settings on the domain controller being used for Identity Manager synchronization for the duration of the migration. Write down the current configuration values and run `ntdsutil.exe` after migration completes to restore the original values. `ntdsutil.exe` can be run on any member server.

1. At a command prompt, type `ntdsutil`.
2. Type `LDAP Policies,` then press Enter.
3. Type `Connections`, then press Enter.
4. Type `Connect to domain` *`domain_name`*, then press Enter.
5. Type `Connect to server` *`server_name`*, then press Enter.
6. Type `Quit`, then press Enter.
7. Type `Show Values`, then press Enter.

```
C:\>ntdsutil
ntdsutil: LDAP Policies
ldap policy: Connections
server connections: Connect to domain raptor
Binding to \\raptor1.raptor.lab ...
Connected to \\raptor1.raptor.lab using credentials of locally logged on user.
server connections: Connect to server raptor1
Disconnecting from \\raptor1.raptor.lab...
Binding to raptor1 ...
Connected to raptor1 using credentials of locally logged on user.
server connections: Quit
ldap policy: Show Values

Policy                          Current(New)
MaxPoolThreads                  4
MaxDatagramRecv                 4096
MaxReceiveBuffer                10485760
InitRecvTimeout                 120
MaxConnections                  5000
MaxConnIdleTime                 900
MaxPageSize                     1000
MaxQueryDuration                120
MaxTempTableSize                10000
MaxResultSetSize                262144
MaxNotificationPerConn          5
MaxValRange                     1500
```

```
ldap policy: set MaxQueryDuration to 1200
ldap policy: set MaxResultSetSize to 6000000
ldap policy: Commit Changes
ldap policy: Quit
ntdsutil: Quit
Disconnecting from raptor1...
C:\>
```

10.14 Error Messages

The following sections contains a list of common error messages.

- "LDAP_SERVER_DOWN" on page 121
- "LDAP_AUTH_UNKNOWN" on page 121
- "Error initializing connection to DirXML: SSL library initialization error: error:00000000:lib(0) :func(0) :reason(0)" on page 122

LDAP_SERVER_DOWN

Source: The status log or DSTrace screen.

Explanation: The driver can't open the LDAP port on the Active Directory domain controller configured for synchronization.

Possible Cause: The server named in the driver authentication context is incorrect.

Possible Cause: You are using an IP address for authentication context, and you have disabled non-Kerberos authentication to Active Directory. Kerberos requires a DNS name for authentication context.

Possible Cause: You have incorrectly configured the driver to use an SSL connection to Active Directory.

Action: The authentication context should hold the DNS name or the IP address of the domain controller you use for synchronization. If you leave the parameter empty, the driver attempts to connect to the machine that is running the driver shim (either the same server that is running IDM, or the server hosting the Remote Loader).

Action: The driver shim can authenticate only using the pre-Windows 2000 Logon method or simple bind. If you have disabled NTLM, NTLM2, and simple bind on your network, you might receive the LDAP_SERVER_DOWN message.

Action: Something is wrong with the certificate that was imported to the driver shim server, or no certificate was imported.

LDAP_AUTH_UNKNOWN

Source: The status log or DSTrace screen.

Explanation: The driver is unable to authenticate to the Active Directory database.

Action: Try to authenticate to the Active Directory database again.

Solution: Unhide the driver parameter of retry-ldap-auth-unknown to allow the driver to retry the authentication when it fails.

1 Open the driver configuration file in the an XML editor.

2. Search for retry-ldap-auth-unknown.
3. Change the hide="true" to hide="false".
4. Access the driver parameters, see Section E.1.5, "Driver Parameters," on page 171 for more information.
5. Select *Driver Settings > Access Options > Retry LDAP Auth unknown* error, then select *Yes*.
6. Click *OK*, then restart the driver.

Error initializing connection to DirXML: SSL library initialization error: error:00000000:lib(0) :func(0) :reason(0)

Source: The status log or DSTrace screen.

Explanation: The Remote Loader cannot make an SSL connection to the Identity Manager engine.

Possible Cause: Incorrect format for the certificate file.

Action: If you are running a Windows 2003 R2 SP1 32-bit server, and are using a self-signed certificate format of DER, the connection fails. The certificate has to have a base 64 format for the SSL connection to work.

10.15 Troubleshooting Driver Processes

Viewing driver processes is necessary to analyze unexpected behavior. To view the driver processing events, use DSTrace. You should only use it during testing and troubleshooting the driver. Running DSTrace while the drivers are in production increases the utilization on the Identity Manager server and can cause events to process very slowly.

10.15.1 Viewing Driver Processes

In order to see the driver processes in DSTrace, values are added to the driver set and the driver objects. You can do this in Designer and iManager.

- "Adding Trace Levels in Designer" on page 122
- "Adding Trace Levels in iManager" on page 124
- "Capturing Driver Processes to a File" on page 125

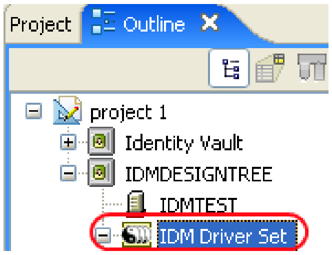
Adding Trace Levels in Designer

You can add trace levels to the driver set object or to each driver object.

- "Driver Set" on page 122
- "Driver" on page 123

Driver Set

1. In an open project in Designer, select the driver set object in the *Outline* view.



2 Right-click and select *Properties*, then click *5. Trace*.

3 Set the parameters for tracing, then click *OK*.

Parameter	Description
Driver trace level	As the driver object trace level increases, the amount of information displayed in DSTrace increases. Trace level 1 shows errors, but not the cause of the errors. If you want to see password synchronization information, set the trace level to 5.
XSL trace level	DSTrace displays XSL events. Set this trace level only when troubleshooting XSL style sheets. If you do not want to see XSL information, set the level to zero.
Java debug port	Allows developers to attach a Java* debugger.
Java trace file	When a value is set in this field, all Java information for the driver set object is written to a file. The value for this field is the path for that file. As long as the file is specified, Java information is written to this file. If you do not need to debug Java, leave this field blank.
Trace file size limit	Allows you to set a limit for the Java trace file. If you set the file size to *Unlimited*, the file grows in size until there is no disk space left. **NOTE:** The trace file is created in multiple files. Identity Manager automatically divides the maximum file size by ten and creates ten separate files. The combined size of these files equals the maximum trace file size.

If you set the trace level on the driver set object, all drivers appear in the DSTrace logs.

Driver

1 In an open project in Designer, select the driver object in the *Outline* view.

2 Right-click and select *Properties*, then click *8. Trace*.

3 Set the parameters for tracing, then click *OK*.

Parameter	Description
Trace level	As the driver object trace level increases, the amount of information displayed in DSTrace increases. Trace level 1 shows errors, but not the cause of the errors. If you want to see password synchronization information, set the trace level to 5. If you select *Use setting from Driver Set*, the value is taken from the driver set object.
Trace file	Specify a filename and location for where the Identity Manager information is written for the selected driver. If you select *Use setting from Driver Set*, the value is taken from the driver set object.
Trace file size limit	Allows you to set a limit for the Java trace file. If you set the file size to *Unlimited*, the file grows in size until there is no disk space left. If you select *Use setting from Driver Set*, the value is taken from the driver set object. **NOTE:** The trace file is created in multiple files. Identity Manager automatically divides the maximum file size by ten and creates ten separate files. The combined size of these files equals the maximum trace file size.
Trace name	The driver trace messages are prepended with the value entered instead of the driver name. Use this option if the driver name is very long.

If you set the parameters only on the driver object, only information for that driver appears in the DSTrace log.

Adding Trace Levels in iManager

You can add trace levels to the driver set object or to each driver object.

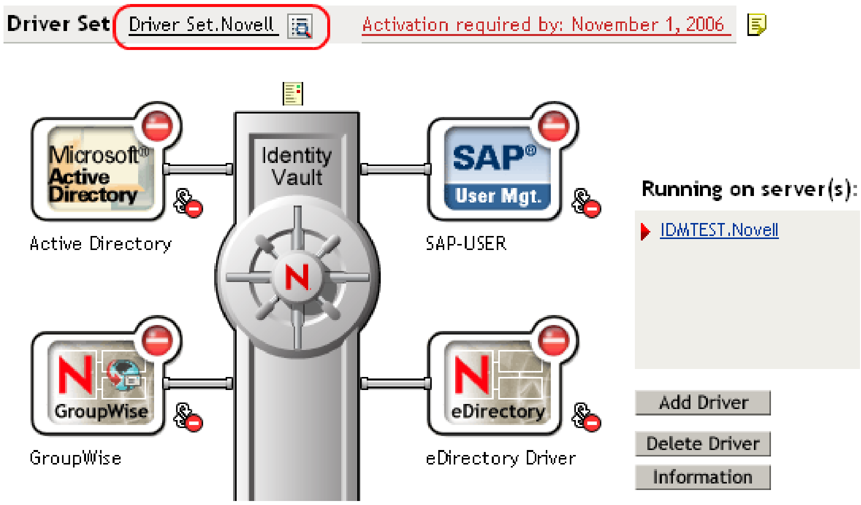
- "Driver Set" on page 124
- "Driver" on page 125

Driver Set

1 In iManager, select *Identity Manager > Identity Manager Overview*.

2 Browse to the driver set object, then click *Search*.

3 Click the driver set name.



4 Select the *Misc* tab for the driver set object.

5 Set the parameters for tracing, then click *OK*.

See "Misc" on page 181 for the parameters.

Driver

1 In iManager, select *Identity Manager > Identity Manager Overview*.

2 Browse to the driver set object where the driver object resides, then click *Search*.

3 Click the upper right corner of the driver object, then click *Edit properties*.

4 Select the *Misc* tab for the driver object.

5 Set the parameters for tracing, then click *OK*.

See "Misc" on page 181 for the parameters.

The option *Use setting from Driver Set* does not exist in iManager.

Capturing Driver Processes to a File

You can save driver processes to a file by using the parameter on the driver object or by using DSTrace. The parameter on the driver object is the *Trace file* parameter, under the *MISC* tab.

The driver processes that are captured through DSTrace are the processes that occur on the Identity Manager engine. If you use the Remote Loader, you need to capture a trace on the Remote Loader at the same time as you are capturing the trace on the Identity Manager engine.

The following methods helps you capture and save Identity Manager processes through DSTrace on different platforms.

- "NetWare" on page 126
- "Windows" on page 126
- "UNIX" on page 126
- "iMonitor" on page 127
- "Remote Loader" on page 127

NetWare

Use `dstrace.nlm` to display trace messages on the system console or trace messages to a file (`sys:\system\dstrace.log`). Use `dstrace.nlm` to display the trace messages to a screen labeled DSTrace Console.

1 Enter `dstrace.nlm` at the server console to load `dstrace.nlm` into memory.

2 Enter `dstrace screen on` at the server console to allow trace messages to appear on the `DSTrace Console` screen.

3 Enter `dstrace file on` at the server console to capture trace messages sent to the DSTrace Console to the `dstrace.log` file.

4 (Optional) Enter `dstrace -all` at the server console to make it easier to read the trace log.

5 Enter `dstrace +dxml dstrace +dvrs` at the server console to display Identity Manager events.

6 Enter `dstrace +tags dstrace +time` at the server console to display message tags and time stamps.

7 Toggle to the DSTrace Console screen and watch for the event to pass.

8 Toggle back to the server console.

9 Enter `dstrace file off` at the server console.

This stops capturing trace messages to the log file. It also stops logging information into the file.

10 Open the `dstrace.log` in a text editor and search for the event or the object you modified.

Windows

1 Open the *Control Panel* > *NDS Services* > `dstrace.dlm`, then click *Start* to display the NDS Server Trace utility window.

2 Click *Edit* > *Options*, then click *Clear All* to clear all of the default flags.

3 Select *DirXML* and *DirXML Drivers*.

4 Click OK.

5 Click *File* > *New*.

6 Specify the filename and location where you want the DSTrace information saved, then click *Open*.

7 Wait for the event to occur.

8 Click *File* > *Close*.

This stops the information from being written to the log file.

9 Open the file in a text editor and search for the event or the object you modified.

UNIX

1 Enter `ndstrace` to start the ndstrace utility.

2 Enter `set ndstrace=nodebug` to turn off all trace flags currently set.

3 Enter `set ndstrace on` to display trace messages to the console.

4 Enter `set ndstrace file on` to capture trace messages to the `ndstrace.log` file in the directory where eDirectory is installed. By default it is `/var/nds`.

5 Enter `set ndstrace=+dxml` to display the Identity Manager events.

6 Enter `set ndstrace=+dvrs` to display the Identity Manager driver events.

7 Wait for the event to occur.

8 Enter `set ndstrace file off` to stop logging information to the file.

9 Enter `exit` to quite the ndstrace utility.

10 Open the file in a text editor. Search for the event or the object that was modified.

iMonitor

iMonitor allows you to get DSTrace information from a Web browser. It does not matter where Identity Manager is running. The following files run iMonitor:

- `ndsimon.nlm` runs on NetWare®.
- `ndsimon.dlm` runs on Windows.
- `ndsimonitor` runs on UNIX.

1 Access iMonitor from http://*server_ip*:8008/nds.

Port 8008 is the default.

2 Specify a username and password with administrative rights, then click *Login*.

3 Select *Trace Configuration* on the left side.

4 Click *Clear All*.

5 Select *DirXML* and *DirXML Drivers*.

6 Click *Trace On*.

7 Select *Trace History* on the left side.

8 Click the document with the *Modification Time* of *Current* to see a live trace.

9 Change the *Refresh Interval* if you want to see information more often.

10 Select *Trace Configuration* on the left side, then click *Trace Off* to turn the tracing off.

11 Select *Trace History* to view the trace history.

The files are distinguished by their time stamp.

If you need a copy of the HTML file, the default location is:

- NetWare: `sys:\system\ndsimon\dstrace*.htm`
- Windows: `*Drive_letter*:\novell\nds\ndsimon\dstrace\*.htm`
- UNIX: `/var/nds/dstrace/*.htm`

Remote Loader

You can capture the events that occur on the machine by running the Remote Loader service.

1 Launch the Remote Loader Console by clicking the icon.

2 Select the driver instance, then click *Edit*.

3 Set the *Trace Level* to 3 or above.

4 Specify a location and file for the trace file.

5. Specify the amount of disk space that the file is allowed.
6. Click *OK*, twice to save the changes.

You can also enable tracing from the command line by using the switches Table 10-2. For more information, see "Configuring the Remote Loader " in the *Novell Identity Manager 3.5.1 Administration Guide*.

Table 10-2 *Command Line Tracing Switches*

Option	Short Name	Parameter	Description
-trace	-t	integer	Specifies the trace level. This is only used when hosting an application shim. Trace levels correspond to those used on the Identity Manager server. Example: `-trace 3` or `-t3`
-tracefile	-tf	filename	Specifies a file to write trace messages to. Trace messages are written to the file if the trace level is greater than zero. Trace messages are written to the file even if the trace window is not open. Example: `-tracefile c:\temp\trace.txt` or `-tf c:\temp\trace.txt`
-tracefilemax	-tfm	size	Specifies the approximate maximum size that trace file data can occupy on disk. If you specify this option, there is a trace file with the name specified using the tracefile option and up to 9 additional "roll-over" files. The roll-over files are named using the base of the main trace filename plus "_n", where n is 1 through 9. The size parameter is the number of bytes. Specify the size by using the suffixes K, M, or G for kilobytes, megabytes, or gigabytes. If the trace file data is larger than the specified maximum when the Remote Loader is started, the trace file data remains larger than the specified maximum until roll-over is completed through all 10 files. Example: `-tracefilemax 1000M` or `-tfm 1000M`

11 Backing Up the Driver

You can use Designer or iManager to create an XML file of the driver. The file contains all of the information entered into the driver during configuration. If the driver becomes corrupted, the exported file can be imported to restore the configuration information.

IMPORTANT: If the driver has been deleted, all of the associations on the objects are purged. When the XML file is imported again, new associations are created through the migration process.

Not all server-specific information stored on the driver is contained in the XML file. Make sure this information is documented through the Doc Gen process in Designer. See "Documenting Projects" in the *Designer 2.1 for Identity Manager 3.5.1*.

- Section 11.1, "Exporting the Driver in Designer," on page 129
- Section 11.2, "Exporting the Driver in iManager," on page 129

11.1 Exporting the Driver in Designer

1. Open a project in Designer, then right-click the driver object.
2. Select *Export to Configuration File*.
3. Specify a unique name for the configuration file, browse to location where it should be saved, then click *Save*.
4. Click *OK* in the Export Configuration Results window.

11.2 Exporting the Driver in iManager

1. In iManager, select *Identity Manager > Identity Manager Overview*.
2. Browse to and select the driver set object, then click *Search*.
3. Click the driver icon.
4. Select *Export* in the Identity Manager Driver Overview window.
5. Browse to and select the driver object you want to export, then click *Next*.
6. Select *Export all policies, linked to the configuration or not* or select *Only export policies that are linked to the configuration*, depending upon the information you want to have stored in the XML file.
7. Click *Next*.
8. Click *Save As*, then click *Save*.
9. Browse and select a location to save the XML file, then click *Save*.
10. Click *Finish*.

Security: Best Practices

12

For more information on how to secure the driver and the information it is synchronizing, see "Security: Best Practices" in the *Novell Identity Manager 3.5.1 Administration Guide*. This section contains a description of the security parameters unique to the Active Directory driver.

During installation, the driver gathers the necessary information and creates default security policies and parameters. Before you begin customizing your Active Directory driver, you should become familiar with the following:

- Default policies and parameters
- The topics discussed in Chapter 10, "Troubleshooting," on page 115, so you can decide whether any of these issues apply to your environment

Understanding how the parameters work together and work with the operating system helps you define your approach to security for Identity Manager data synchronization.

- Section 12.1, "Default Configuration of the Security Parameters," on page 131
- Section 12.2, "Recommended Security Configurations when Using the Remote Loader," on page 133
- Section 12.3, "Recommended Security Configurations when Using the Simple Authentication Method," on page 134

12.1 Default Configuration of the Security Parameters

The security parameters must be changed after the initial configuration of driver occurs.

To change these parameters in iManager:

1. Click *Identity Manager > Identity Manager Overview*, then click *Search* to search for the driver set that is associated with the driver.
2. Browse to the driver, then click the upper right corner of the driver icon.
3. Click *Edit Properties > Driver Configuration > Driver Parameters*.
4. Review the driver parameters in Table 12-1, and decide if you need to make any changes.

To change these parameters in Designer:

1. Open a project in the modeler, then right-click the driver line and select *Properties > Driver Configuration*.
2. Click *Driver Parameters*.
3. Review the driver parameters in Table 12-1, and decide if you need to make any changes.

Table 12-1 *Security Parameters*

Security Parameter	Description
Authentication ID	The account the driver uses to access the domain data. The *Authentication ID* can be specified using different formats: ◆ If the *Authentication method* is set to *negotiate*, the user name is specified with the domain name or the full qualified domain name. For example, `user` or `domain\user`. ◆ If the *Authentication method* is set to *simple*, the user name must be specified using an LDAP fully distinguished name. For example, `cn=IDMadmin,cn=Users,dc=domain,dc=com`.
Authentication context	The context used to access domain data. The *Authentication context* can be specified using different formats: ◆ If the *Authentication method* is set to *negotiate*, use the DNS name of the Active Directory domain controller. For example, `mycontroller.mydomain.com`. ◆ If the *Authentication method* is set to *simple*, use the DNS name of the Active Directory domain controller or the IP address of the LDAP server. For example, `mycontroller.mydomain.com` or `10.0.0.1`.
Application password	The password for the *Authentication ID* account.
Authentication Method	The method of authentication to Active Directory. *Negotiate* uses Microsoft's security package to negotiate the logon type. Typically Kerberos or NTLM is selected. *Simple* uses LDAP style simple bind for logon. If you want to use Password Synchronization, select *Negotiate*.
Digitally sign communications	This setting requires Windows 2003 or Windows 2000 with the most recent support pack, and Internet Explorer 5.5 SP2 or later on both servers. This enables signing on a Kerberos or NTLM v2 authenticated connection. Select *Yes* to digitally sign the communication between the driver shim and Active Directory. This does not hide the data from view on the network, but it reduces the chance of security attacks. Signing only works when you use the *Negotiate* authentication method and the underlying security provider selects NTLM v2 or Kerberos for its protocol. Do not use this option with SSL. Select *No* to have communications not signed.

Security Parameter	Description
Digitally sign and seal communications	This setting requires Windows 2003 or Windows 2000 with the most recent support pack, and Internet Explorer 5.5 SP2 or later on both servers. This setting enables encryption on a Kerberos or NTLM v2 authenticated connection. Select *Yes* to digitally encrypt communication between the driver shim and the Active Directory database. Sealing only works when you use the *Negotiate* authentication method and the underlying security provider selects NTLM v2 or Kerberos for its protocols. Do not use this option with SSL. Select *No* to not have communication between the driver shim and the Active Directory database signed and sealed.
Use SSL for encryption	Select *Yes* to digitally encrypt communication between the driver shim and the Active Directory database. This option can be used with *Negotiate* or *Simple* authentication methods. SSL requires that the Microsoft server running the driver shim imports the domain controller's server certificate. For more information, see Securing Windows 200 Server (http://www.microsoft.com/technet/security/prodtech/windows2000/secwin2k/swin2kad.mspx). By default, the parameter is set to *No*. If you set this value to *Yes*, the SSL pipe is encrypted for the entire conversation. An encrypted pipe is preferred because the driver typically synchronizes sensitive information. However, encryption slows the general performance of your servers.
Logon and impersonate	Select *Yes* to logon and impersonate the driver authentication account for CDOEXM (Collaboration Data Object for Exchange Management) and Password Set support. The driver performs a local logon. The authentication account must have the proper rights assignment. For more information, see Section 2.4, "Creating an Administrative Account," on page 28. If *No* is selected, the driver performs a network logon only.

12.2 Recommended Security Configurations when Using the Remote Loader

If you are using the Remote Loader, the following table lists the recommended security configurations for the driver.

Table 12-2 *Recommended Security Configuration for the Remote Loader*

Parameter	Description
Authentication ID	The account the driver uses to access the domain data. Use the domain logon name, for example Administrator.

Parameter	Description
Authentication Context	The DNS name of the domain controller. If you don't want to run the driver on your Active Directory domain controller, use *hostname* for the Negotiate method but use *hostname* or the IP address for the *simple* method.
Application Password	The password used for the *Authentication ID*.
Remote Loader Password	The password for the Remote Loader service.
Authentication Method	Select *negotiate*.
Digitally sign communications	Select *No*. Requires Windows 2003 or Windows 2000 with the most recent support pack, and Internet Explorer 5.5 SP2 or later on both servers.
Digitally sign and seal communications	Select *No*. Requires Windows 2003 or Windows 2000 with the most recent support pack, and Internet Explorer 5.5 SP2 or later on both servers.
Use SSL for encryption	Select *Yes*. SSL is required to perform a Subscriber password check, a Subscriber password set, and a Subscriber password modify when the driver shim isn't running on the domain controller.

12.3 Recommended Security Configurations when Using the Simple Authentication Method

SSL is recommended if you have selected the Simple authentication mechanism because Simple authentication passes passwords in clear text.

Table 12-3 *Recommended Security Configuration when Using the Simple Authentication Method*

Parameter	Description
Authentication ID	The account the driver uses to access the domain data. Use LDAP format for the *Authentication ID*. For example, cn=IDMadmin,cn=Users,dc=domain,dc=com
Authentication Context	IP address of domain controller.
Password	The password for the specified *Authentication ID*.
Digitally sign communications	Select *No*.
Digitally sign and seal communications	Select *No*.
Use SSL for encryption	Select *Yes*. SSL requires that the Microsoft server running the driver shim imports the domain controller's server certificate imported. For more information, see Securing Windows 200 Server (http://www.microsoft.com/technet/security/prodtech/windows2000/secwin2k/swin2kad.mspx).

Changing Permissions on the CN=Deleted Objects Container

A

When an Active Directory object is deleted, a small portion of the object remains for a specified time so that other domain controllers that are replicating changes become aware of the deletion. By default, only the System account and members of the Administrators group can view the contents of this container. This section describes how to modify the permissions on the CN=Deleted Objects container.

Changing permissions on the Deleted Objects container might be necessary if you have enterprise applications or services that bind to Active Directory with a non-System or non-Admin account and poll for directory changes.

This process requires `dscals.exe` from the Active Directory Application Mode (ADAM) package. This version is an upgrade from the one in the Windows Server 2003 Support Tools and now supports the required capabilities. The ADAM Administration Tools are supported on Windows XP Professional, Windows Server 2003 Standard Edition, Windows Server 2003 Enterprise Edition, and Windows Server 2003 Datacenter Edition.

To get and install the ADAM Administration Tools:

1. From the ADAM Web page (http://www.microsoft.com/downloads/details.aspx?FamilyId=9688F8B9-1034-4EF6-A3E5-2A2A57B5C8E4&displaylang=en), download the ADAM retail package.
2. Double-click the downloaded file and provide a directory to extract the archive into.
3. Launch the Active Directory Application Mode Setup Wizard by double-clicking `adamsetup.exe`, then click *Next*.
4. Review and accept the license terms, then click *Next*.
5. Select ADAM administration tools only, then click *Next*.
6. Review the selections, then click *Next*.
7. When Setup has concluded, click *Finish*.

After the ADAM Administration Tools are installed, modify the permissions on the CN=Deleted Objects container:

1. Log in with a user account that is a member of the Domain Admins group.
2. *Click Start > All Programs > ADAM > ADAM Tools Command Prompt*.
3. At the command prompt, enter the following command:

   ```
   dsacls "CN=Deleted Objects,DC=Contoso,DC=com" /takeownership
   ```

   Substitute the distinguished name of the Deleted Objects container for your own domain.

   Each domain in the forest will have its own Deleted Objects container.

   The following output should be displayed:

```
Owner: Contoso\Domain Admins
  Group: NT AUTHORITY\SYSTEM
  Access list:
  {This object is protected from inheriting permissions from the parent}
  Allow BUILTIN\Administrators  SPECIAL ACCESS
                              LIST CONTENTS
                              READ PROPERTY
  Allow NT AUTHORITY\SYSTEM     SPECIAL ACCESS
                              DELETE
                              READ PERMISSONS
                              WRITE PERMISSIONS
                              CHANGE OWNERSHIP
                              CREATE CHILD
                              DELETE CHILD
                              LIST CONTENTS
                              WRITE SELF
                              WRITE PROPERTY
                              READ PROPERTY
The command completed successfully
```

4. To grant a security principal permission to view the objects in the CN=Deleted Objects container, enter the following command:

```
dsacls "CN=Deleted Objects,DC=Contoso,DC=com" /g CONTOSO\JaneDoe:LCRP
```

In this example, the user CONTOSO\JaneDoe has been granted List Contents and Read Property permissions on the container. These permissions are sufficient to allow the user to view the contents of the Deleted Objects container. However, these permissions don’t allow the user to make any changes to objects in that container. These permissions are equivalent to the default permissions granted to the Administrators group. By default, only the System account has permission to modify objects in the Deleted Objects container.

The following output should be displayed:

```
  Owner: CONTOSO\Domain Admins
Group: NT AUTHORITY\SYSTEM
  Access list:
  {This object is protected from inheriting permissions from the parent}
  Allow BUILTIN\Administrators  SPECIAL ACCESS
                              LIST CONTENTS
                              READ PROPERTY
Allow NT AUTHORITY\SYSTEM     SPECIAL ACCESS
                              DELETE
                              READ PERMISSONS
                              WRITE PERMISSIONS
                              CHANGE OWNERSHIP
                              CREATE CHILD
                              DELETE CHILD
                              LIST CONTENTS
                              WRITE SELF
                              WRITE PROPERTY
                              READ PROPERTY
Allow CONTOSO\JaneDoe         SPECIAL ACCESS
                              LIST CONTENTS
                              READ PROPERTY
The command completed successfully.
```

The user CONTOSO\JaneDoe now has permissions to view deleted objects in the CONTOSO domain.

Configuring for Use with an ADAM Instance

B

The Active Directory driver can be configured for use with a Microsoft Active Directory Application Mode (ADAM) instance. You import a configuration file to create a driver to connect to the ADAM instance.

There are multiple ways to configure your environment to synchronize the information. For example, Novell® recommends setting up your own certification authority (CA) in order to issue certificates that can be used for SSL connections to ADAM. If you already have server certificates, or if you have access to another CA that can issue valid certificates, you can ignore the steps that describe how to set up your own CA. Likewise, if you don't want to configure SSL (required if you want to set passwords on the Subscriber channel) then you can skip the entire section about configuring Certificate Services.

Any discussion of setting passwords is referring to the Subscriber channel (from IDM to ADAM). Password synchronization on the Publisher channel (from ADAM to IDM) is not currently possible, unless a regular user attribute (not the userPassword attribute) is used in ADAM to store the password.

- Section B.1, "Prerequisites," on page 137
- Section B.2, "Installation Tasks," on page 137
- Section B.3, "Configuration Tasks," on page 139

B.1 Prerequisites

To achieve synchronization with an ADAM instance, you need the following items installed on one or more computers running Windows 2003 Server. Windows 2003 is the only supported platform for this configuration.

- An Identity Manager server or Remote Loader where the Active Directory driver is configured.
- Internet Information Services (IIS) (must be installed before Certificate Services)
- Certificate Services
- A certification authority (can be your own standalone CA configured when you install Certificate Services)
- An ADAM instance (this example uses a standalone instance)

B.2 Installation Tasks

The following installation tasks must be completed in the order that they are listed. If you do not need to do a step, you can skip it.

- Section B.2.1, "Installing Internet Information Services," on page 138
- Section B.2.2, "Installing Certificate Services," on page 138
- Section B.2.3, "Installing ADAM," on page 138

- Section B.2.4, “Requesting and Installing the Server Certificate,” on page 139
- Section B.2.5, “Installing Identity Manager,” on page 139

B.2.1 Installing Internet Information Services

If you want to set up your own CA in order to configure SSL on ADAM, you need to install Internet Information Services (IIS).

1. On your Windows 2003 Server, access the Control Panel, then click *Add or Remove Programs*.
2. In the left pane, select *Add/Remove Windows Components*.
3. Select *Application Server*, then click *Details*.
4. Select *Internet Information Services (IIS)*, then click *Details*.
5. Verify that at least the following are selected:
   - *Common Files*
   - *Internet Information Services Manager*
   - *World Wide Web Service*
6. Click *OK* twice, then click *Next* to complete the installation.

   You might be prompted to insert your original installation media for Windows 2003 Server.

B.2.2 Installing Certificate Services

1. On your Windows 2003 Server, access the Control Panel, then click *Add or Remove Programs*.
2. In the left pane, select *Add/Remove Windows Components*.
3. Select to *Certificate Services*, then click *Next* to complete the installation.

B.2.3 Installing ADAM

1. On your Windows 2003 Server, access the Control Panel, then click *Add or Remove Programs*.
2. In the left pane, select *Add/Remove Windows Components*.
3. Select *Active Directory Services*, then click *Details*.
4. Put a check mark next to *Active Directory Application Mode (ADAM)*, then click *OK*.
5. Click *Next* to complete the installation.

   The AD driver doesn't currently have a way to change the port when making a connection, so you need to use the defaults. If the values default to something else, you probably already have a service using those ports, and you might need to disable or uninstall the other service.
6. Click *Next*.
7. Select *Yes* to create an application directory partition, unless you plan on doing it later.
8. Specify the DN of the location where you'd like to synchronize users. For example, `CN=People,DC=adamtest1,DC=COM`.
9. Click *Next*.
10. Leave the default locations for data files and data recovery files, then click *Next*.
11. Choose an account for the ADAM service, then click *Next*.

NOTE: If you are installing ADAM on a server that is not already part of a domain, you might get a warning at this point. This is usually not a problem with ADAM, and you should continue with the installation.

12 Click *Next* to assign the current user (the one you are logged in as) rights to administrate ADAM.

13 Select *Import the selected LDIF files for this instance of ADAM*.

14 Select *MS-User.LDF*, then click *Add*.

15 Click *Next*.

16 Review the installation summary, then click *Next*.

B.2.4 Requesting and Installing the Server Certificate

1 On the server where you installed IIS and Certificate Services, specify the following address in a Web browser: `http://localhost/certsrv`.

2 You should see a welcome message from Certificate Services. If you do not, go back and make sure you have IIS and Certificate Services both installed.

3 The steps for requesting and installing a certificate are found at [.NET] Using SSL with ADAM (http://erlend.oftedal.no/blog/?blogid=7).

4 On your ADAM server, make sure you have the certificate installed in the following location in MMC: Certificates - Service (adaminstance) on Local Computer\ADAM_adaminstance\Personal.

5 On the Identity Manager server (or the Remote Loader computer) where the driver is running, you need the CA certificate only and it must be in Certificates - Current User\Trusted Root Certificates.

See Active Directory Application Mode: Frequently Asked Questions (http://www.microsoft.com/windowsserver2003/adam/ADAMfaq.mspx) for additional resources.

B.2.5 Installing Identity Manager

Identity Manager must be installed to use the ADAM driver. To install Identity Manager, see the *Identity Manager 3.5.1 Installation Guide*. If you are going to use the Remote Loader, see “Deciding Whether to Use the Remote Loader”.

B.3 Configuration Tasks

- Section B.3.1, “Setting the Default Naming Context for your ADAM Instance,” on page 139
- Section B.3.2, “Creating a User in ADAM with Sufficient Rights,” on page 140
- Section B.3.3, “Creating the ADAM Driver,” on page 140

B.3.1 Setting the Default Naming Context for your ADAM Instance

1 Start the ADSI Edit application by selecting *Start* > *All Programs* > *ADAM* > *ADAM ADSI Edit*.

2 In the tree view, select the root item called *ADAM ADSI Edit*.

3 Under the *Action* menu, select *Connect to*.

4 In the *Connection name* field, type `Configuration`.

5 Select *Well-known naming context*. Make sure the value in the drop-down list is set to *Configuration*.

6 Set the other authentication credentials as appropriate, then click *OK*.

7 In the tree view, expand the *Configuration* item and those items underneath it until you can select the following entry:

```
CN=NTDS Settings,CN=ServerName$InstanceName,CN=Servers,
CN=Default-First-Site-Name, CN=Sites,CN=Configuration,CN={GUID}
```

Keep in mind that in the above DN, you should replace ServerName, InstanceName, and GUID with those values actually used in your ADAM instance.

8 Under the *Action* menu, select *Properties*.

9 Select the *msDS-DefaultNamingContext* attribute, then click *Edit*.

10 Specify the same value you used in Step 8 in Section B.2.3, "Installing ADAM," on page 138.

11 Click *OK* twice.

12 Restart your ADAM instance so the new default naming context takes effect.

B.3.2 Creating a User in ADAM with Sufficient Rights

For the driver to work properly, it is best to create a user object specifically for the driver to use. This user should only have the rights to do the work that is required. For more information see, Section 2.4, "Creating an Administrative Account," on page 28.

B.3.3 Creating the ADAM Driver

You can create the ADAM driver through Designer or iManager.

- "Creating the ADAM Driver in Designer" on page 140
- "Creating the ADAM Driver in iManager" on page 140

Creating the ADAM Driver in Designer

1 Open a project in Designer. In the Modeler, right-click the driver set and select *New* > *Driver*.

2 From the drop-down list, select *ADAM,* then click *Run*.

3 Configure the driver by filling in the fields. Specify information for your environment. For information on the settings, see Table B-1 on page 141 for more information.

4 After specifying parameters, click *Finish* to import the driver.

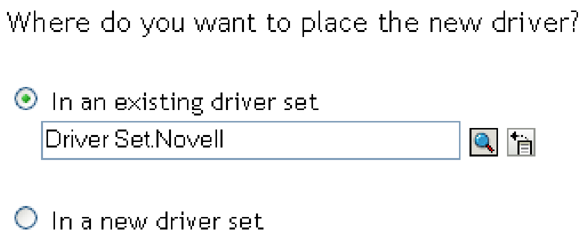
5 After the driver is imported, customize and test the driver.

6 After the driver is fully tested, deploy the driver into the Identity Vault. See "Deploying a Driver to an Identity Vault" in the *Designer 2.1 for Identity Manager 3.5.1*.

Creating the ADAM Driver in iManager

1 In iManager, select *Identity Manager Utilities* > *Import Configuration*.

2 Select a driver set, then click *Next*.



Where do you want to place the new driver?

● In an existing driver set

Driver Set.Novell

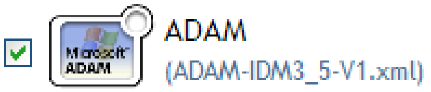
○ In a new driver set

If you place this driver in a new driver set, you must specify a driver set name, context, and associated server.

3 Select how you want the driver configurations sorted:

- All configurations
- Identity Manager 3.5 configurations
- Identity Manager 3.0 configurations
- Configurations not associated with an IDM version

4 Select the *ADAM* driver, then click *Next*.



☑ ADAM
(ADAM-IDM3_5-V1.xml)

5 Configure the driver by filling in the configuration parameters, then click *Next*. For information on the settings, see Table B-1 on page 141.

6 Define security equivalences, using a user object that has the rights that the driver needs to have on the server, then click *OK*.

Use the user created in Section B.3.2, "Creating a User in ADAM with Sufficient Rights," on page 140.

7 Identify all objects that represent administrative roles and exclude them from replication, then click *OK*.

Exclude the security-equivalence object (for example, DriversUser) that you specified in Step 6. If you delete the security-equivalence object, you have removed the rights from the driver, and the driver can't make changes to Identity Manager.

8 Click *Finish*.

NOTE: The parameters are presented on multiple screens. Some parameters are only displayed if the answer to a previous prompt requires more information to properly configure the policy.

Table B-1 *Configuration Parameters for the ADAM Driver*

Parameter	Description
Driver name	Specify the name of the driver object.

Parameter	Description
Authentication ID	Specify the name of the user object created in Section B.3.2, “Creating a User in ADAM with Sufficient Rights,” on page 140. The name needs to be specified as a full LDAP DN. Example, `CN=IDM,CN=Users,DC=domain,DC=com`
Authentication Password	Specify the password of the user object with sufficient rights.
Authentication Context	Specify the DNS name or IP address of the ADAM instance server.
Driver Polling Interval	Specify the number of minutes to delay before querying Active Directory for changes. The default value is 1 minute.
Password Sync Timeout	Specify the number of minutes for the driver to attempt to synchronize a given password. The default value is 5 minutes.
Driver is Local/Remote	Configure the driver for use with the Remote Loader service by selecting *Remote*, or select *Local* to configure the driver for local use. For more information, see “Deciding Whether to Use the Remote Loader” in the *Novell Identity Manager 3.5.1 Administration Guide*.
Name mapping policy selection	Select whether to accept the full policy or parts of the policy manually. The policy maps the Identity Vault Full Name attribute to the Active Directory object name and the policy maps the Active Directory Pre-Windows 2000 Logon Name to the Identity Vault user name.
Remote Host Name and Port	Remote option only. The host name or IP address and port number where the Remote Loader Service has been installed and is running for this driver. The default port is 8090. This setting displays only if you set *Driver is Local/Remote* to *Remote*.
Driver Password	Remote option only. The Remote Loader uses the Driver Object Password to authenticate itself to the Identity Manager server. The password must be the same password that is specified as the Driver object password on the Remote Loader. This setting displays only if you set *Driver is Local/Remote* to *Remote*.
Remote Password	Remote option only. The Remote Loader password is used to control access to the Remote Loader instance. The password must be the same password that is specified as the Remote Loader password on the Remote Loader. This setting displays only if you set *Driver is Local/Remote* to *Remote*.

Parameter	Description
Full Name Mapping	Name mapping policy selection only. Select *Yes* if you want the Identity Vault Full Name attribute to be synchronized with the Active Directory object name and display name. This policy is useful when creating user accounts in Active Directory by using the Microsoft Management Console Users and Computer snap-in.
Logon Name Mapping	Select *Yes* if you want the Identity Vault object name synchronized with the Active Directory Pre-Windows 2000 Logon Name (also known as the NT Logon Name and the sAMAccountName). This policy is useful when creating user accounts in Active Directory by using the Microsoft Management Console Users and Computer snap-in.
Import will proceed to Active Directory logon name policy selections	Select *OK*.
Base container in eDirectory	Specify the base container in the Identity Vault for synchronization. This container is used in the Subscriber Matching policies to limit the Identity Vault objects being synchronized and in the Publisher Placement policies when adding objects to the Identity Vault. New users are placed in this container by default. Use the dot format. For example, `users.myorg.`
Publisher Placement	*Mirrored* places objects hierarchically within the base container. *Flat* places objects strictly within the base container. This selection builds the default Publisher Placement policies. **NOTE:** If you select *Mirrored*, the driver assumes the structure of the eDirectory™ database is the same in Active Directory as the eDirectory base container. If the structure is not the same, the objects are not placed properly. Create the same structure in Active Directory that exists in eDirectory, or migrate the eDirectory containers before migrating User objects.

Base container in Active Directory	Specify the base container in Active Directory, in LDAP format. New users are placed in this container by default. For example,
	`CN=Users,DC=MyDomain,DC=com`
	If the target container doesn't exist, you must create it and make sure it is associated with the eDirectory base container before trying to add users to this container.
	If you are creating or using a container other than Users in Active Directory, the container is an OU, not a CN. For example,
	`OU=Sales,OU=South,DC=MyDomain,DC=com`

Parameter	Description
Active Directory Placement	*Mirrored* places the objects hierarchically within the base container. *Flat* places objects strictly within the base container. This selection builds the default Subscriber Placement policies. **NOTE:** If you select *Mirrored*, the driver assumes the structure of the Active Directory database is the same in eDirectory as the Active Directory base container. If the structure is not the same, the objects are not placed properly. Create the same structure in eDirectory that exists in Active Directory, or migrate the Active Directory containers before migrating User objects.
Configure Data Flow	Establishes the initial driver filter that controls the classes and attributes that will be synchronized. The purpose of this option is to configure the driver to best express your general data flow policy. It can be changed after import to reflect specific requirements. *Bidirectional* sets classes and attributes to synchronize on both the Publisher and Subscriber channels. A change in either the Identity Vault or Active Directory is reflected on the other side. Use this option if you want both sides to be authoritative sources of data. *AD to Vault* sets class and attributes to synchronize on the Publisher channel only. A change in Active Directory is reflected in the Identity Vault, but Identity Vault changes are ignored. Use this option if you want Active Directory to be the authoritative source of data. *Vault to AD* sets classes and attributes to synchronize on the Subscriber channel only. A change in the Identity Vault is reflected in Active Directory, but Active Directory changes are ignored. Use this option if you want the Identity vault to be the authoritative source of data. **IMPORTANT:** Delete, Move, and Rename events are independent of the filter. It does not matter which option you select, these events are processed by the driver. If you do not want these events to synchronize, you must change the default configuration of the driver. You can use one of the predefined policies that comes with Identity Manager 3.5.1 to change Delete events into Remove Association events. For more information, see “Command Transformation - Publisher Delete to Disable ” in the *Policies in Designer 2.1*. To block Move and Rename events, you must customize the driver.

Parameter	Description
Password Failure Notification User	Password synchronization policies are configured to send e-mail notifications to the associated user when password updates fail. You have the option of sending a copy of the notification e-mail to another user, such as a security administrator. If you want to send a copy, enter or browse for the DN of that user. Otherwise, leave this field blank.
Group membership policy	Configure Elements option only. Group membership in Active Directory can be controlled by synchronizing the membership list or by using Entitlements. *Entitlements* uses the Workflow service or the Role-Based Entitlements to assign group membership. *Synchronize* uses policies to synchronize the group membership list. *None* does not synchronize group membership information.
User Principal Name Mapping	Allows you to choose a method for managing the Active Directory Windows 2000 Logon Name (also known as the userPrincipalName). userPrincipalName takes the form of an e-mail address, such as in usere@domain.com. Although the shim can place any value into userPrincipalName, it is not useful as a logon name unless the domain is configured to accept the domain name used with the name. *Follow Active Directory e-mail address* sets userPrincipalName to the value of the Active Directory mail attribute. This option is useful when you want the user's e-mail address to be used for authentication and Active Directory is authoritative for e-mail addresses. *Follow Identity Vault e-mail address* sets userPrincipalName to the value of the Identity Vault e-mail address attribute. This option is useful when you want the user's e-mail address to be used for authentication and the Identity Vault is authoritative for e-mail addresses. *Follow Identity Vault name* is useful when you want to generate userPrincipalName from the user logon name plus a hard-coded string defined in the policy. *None* is useful when you do not want to control userPrincipalName or when you want to implement your own policy.

Provisioning Exchange Accounts

C

The Active Directory driver can be configured to provision Active Directory accounts as well as Exchange 2000, Exchange 2003, and Exchange 2007 accounts.

The driver can synchronize Exchange 2000 and Exchange 2003 accounts or Exchange 2007 accounts. It cannot synchronize all types of Exchange accounts at the same time. If you have multiple types of Exchange accounts you must set up two separate drivers to synchronize the Exchange 2007 accounts and Exchange 2000 and 2003 accounts.

- Section C.1, "Provisioning Exchange 2000 and 2003 Accounts," on page 147
- Section C.2, "Provisioning Exchange 2007 Accounts," on page 149

C.1 Provisioning Exchange 2000 and 2003 Accounts

There are two different ways to provision the Exchange 2000 and 2003 mailbox accounts with the Active Directory driver. You can set attributes on User objects so a Microsoft program (the Recipient Update Service) can use this information to provision to users to the Exchange database. Or you enable Collaboration Data Objects for Exchange Management (CDOEXM), which is the method documented in this section.

With CDOEXM enabled, an Exchange 2000 or 2003 mailbox is provisioned by setting the homeMDB attribute. When the homeMDB attribute is set, the driver automatically sets all required attributes. The driver can create, delete, and move mailboxes. The mailbox moves that are supported are only interdomain moves.

CDOEXM is an API that is provided by Microsoft. The Active Directory driver uses this API to provision the Exchange accounts.

The homeMDB attribute is set during initial configuration, but you can change the setting by modifying the driver policy. To find out what the homeMDB attribute is for your Exchange system, see Section 5.1, "Using the Active Directory Discovery Tool," on page 49.

To configure the driver to synchronize Exchange 2000 and 2003 accounts:

1. If the server that is running the driver is a non-Exchange server, the Exchange Management tools must be installed on this server.
2. Verify that the authentication account for the driver has enough rights to create, delete, or move Exchange accounts.
3. If the driver is running on a member server, you must use SSL and you must run the Remote Loader service as a specific domain user with enough rights to delete, create, or move Exchange accounts.
4. Run the Active Directory Discovery tool to find out what the homeMDB attribute is for the Exchange 2000 or 2003 system. For more information, see Section 5.1, "Using the Active Directory Discovery Tool," on page 49.

5 Specify the configuration parameters to provision the Exchange mailboxes, when you are creating a driver object. See Table C-1 for a list of Exchange parameters. See Chapter 5, "Configuring the Active Directory Driver," on page 49 for information on how to create the driver object.

6 Verify that you have selected *use-cdoexm* to provision the Exchange 2000 and 2003 mailboxes. See Exchange Management interface type for more information.

Table C-1 *Exchange Provisioning Configuration Parameters*

Parameter	Description
Exchange Policy	Exchange provisioning can be handled by a driver policy, Entitlements, or skipped entirely. A user can be assigned a mailbox in Exchange (the user is mailbox enabled) or have information about a foreign mailbox stored in the Identity Vault record (the user is mail enabled). When using Entitlements, an external service such as the Workflow service or Role-Based Entitlements makes these decisions and the driver policy simply applies them. *Implement in policy* uses the policies in the driver instead of Entitlements to assign Exchange mailboxes. When using the driver policy, the decision to mailbox enable or mail enable a user, plus the Exchange message database where the account will reside, is controlled completely in the policy. When *None* is selected, the default configuration does not create Exchange mailboxes but does synchronize the Identity Vault Internet E-Mail Address with the Active Directory mail attribute.
Exchange Management interface type	The driver cannot provision both Exchange 2007 mailboxes and Exchange 2000 and 2003 mailboxes. This option allows you to select which type of mailboxes the driver can provision. *use-cdoexm* synchronizes Exchange 2000 and Exchange 2003 accounts. *use-post-cdoexm* synchronizes Exchange 2007 accounts.
Allow Exchange mailbox move (yes/no)	When this option is enabled, the driver shim intercepts modifications to the Active Directory homeMDB attribute to move the mailbox to the new message data store. *Yes* moves the Exchange mailbox. *No* does not move the Exchange mailbox.
Allow Exchange mailbox delete (yes/no)	When this option is enabled, the driver shim intercepts removal for the Active Directory homeMDB attribute to delete the mailbox. *Yes* allows the Exchange mailbox to be deleted. *No* does not allow the Exchange mailbox to be deleted.

Parameter	Description
Default Exchange MDB	Specify the default Exchange Message Database (MDB). To obtain the correct name for the Exchange MDB, see Section 5.1, "Using the Active Directory Discovery Tool," on page 49. For example, `[CN=Mailbox Store (CONTROLLER),CN=First Storage Group,CN=InformationStore,CN=CONTROLLER,CN=Servers,CN=First Administrative Group,CN=Administrative Groups,CN=Domain,CN=Microsoft Exchange,CN=Services,CN=Configuration,DC=Domain,DC=com]` The driver can be updated to manage additional MDBs after the import is complete.

C.2 Provisioning Exchange 2007 Accounts

Exchange 20007 no longer supports CDOEXM for mailbox management. In order to provision the Exchange 2007 mailboxes, the Active Directory driver uses Windows PowerShell* in the form of a service.

This service is installed on the server that is running the Active Directory driver. If you decided to run the driver locally, the driver is installed on the Identity Manager server. If you decided to run the driver remotely, the driver is installed on the same server as the Remote Loader service.

The service listens on a default port of 8097. This is set when the service is installed. It is stored in the registry key `HKEY_LOCAL_MACHINE\SOFTWARE\Novell\IDM_AD_EX_SERVICE`. The value can be edited if necessary. If you edit the registry key, both the service and the driver must be restarted.

The Active Directory driver creates, moves, and disables Exchange 2007 mailboxes.

To provisions the Exchange 2007 mailboxes, the following steps must be completed:

1. Meet the prerequisites
2. Install the service
3. Configure the driver

Prerequisites

On the server where the driver will run, whether that is as a Remote Loader service or if the driver is installed locally, the following items must be installed:

- ❑ Microsoft .NET Framework version 2.0 or above.
- ❑ Exchange 2007 Management Tools for the correct platform: 32-bit or 64-bit.

Installing the Service

To install the service, you must use the .NET Framework install utility called `InstallUtil.exe`. The version folder is the current version of the .NET Framework that is installed.

The default location for a 32-bit server is `C:\WINDOWS\Microsoft.Net\Framework\`*`version`*`\InstallUtil.exe`.

The default location for a 64-bit server is `C:\WINDOWS\Microsoft.Net\Framework64\`*`version`*`\InstallUtil.exe`.

To use `InstallUtil.exe`:

1. Open a .NET command prompt.
2. Issue the command `InstallUtil IDMExService.exe` to register the service and create the correct registry entries.
3. To start the service, select *Start* > *Control Panel* > *Administrative Tools* > *Services*.
4. Right-click the service *IDM_AD_Ex2007_Service*, then select *Start*.

To uninstall the service, issue the command `InstallUtil /u IDMExService.exe`.

Configuring the Driver

You need to create a new driver object and select the correct fields to enable provisioning with Exchange 2007 or modify the existing driver.

To create a new driver:

1. Specify the configuration parameters to provision the Exchange 2007 mailboxes, when you are creating a driver object. See Table C-1 for a list of Exchange parameters. See Chapter 5, "Configuring the Active Directory Driver," on page 49 for information on how to create the driver object.
2. Verify you have selected *use-post-cdoexm* to provision Exchange 2007 mailboxes. See "Exchange Management interface type" on page 148 for more information.
3. Start the driver to provision the Exchange 2007 mailboxes.

To modify an existing driver in Designer:

1. Right-click the Active Directory driver in the Modeler, then select *Properties*.
2. Select *Driver Configuration* > *Driver Parameters* > *Edit XML*.
3. Search for the heading `<header display-name="Exchange Options"/>`.
4. Change the following lines:

Old XML	New XML
`<definition display-name="Use CDOEXM for Exchange (yes/no)" name="use-CDOEXM" type="enum">`	`<definition display-name="Exchange Management interface type (use-cdoexm/use-post-cdoexm)" name="exch-api-type" type="enum">`
`<enum-choice display-name="Yes">yes</enum-choice>`	`<enum-choice display-name="use-cdoexm">use-cdoexm</enum-choice>`
`<enum-choice display-name="No">no</enum-choice>`	`<enum-choice display-name="use-post-cdoexm">use-post-cdoexm</enum-choice>`
`<definition display-name="Allow CDOEXM Exchange mailbox move (yes/no)" name="cdoexm-move" type="enum">`	`<definition display-name="Allow Exchange mailbox move (yes/no)" name="exch-move" type="enum">`
`<definition display-name="Allow CDOEXM Exchange mailbox delete (yes/no)" name="cdoexm-delete" type="enum">`	`<definition display-name="Allow Exchange mailbox delete (yes/no)" name="exch-delete" type="enum">`

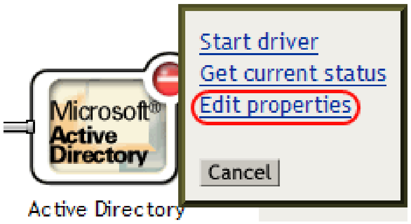
5 Click *OK* twice to save the changes.

To modify an existing driver in iManager:

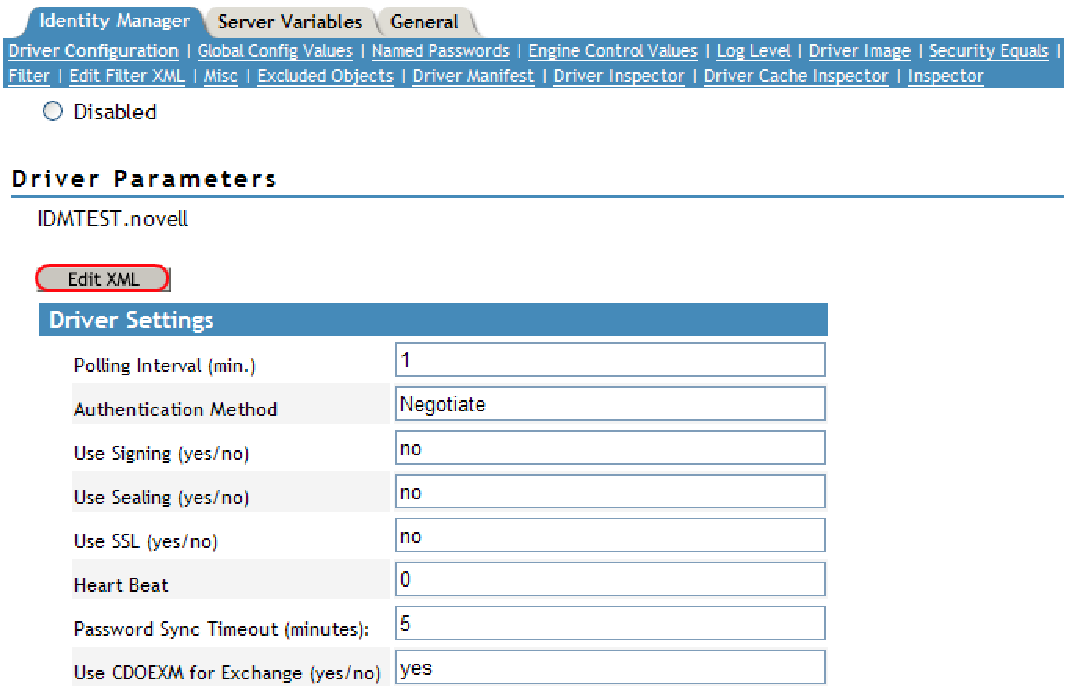
1 Select *Identity Manager > Identity Manager Overview*.

2 Select the driver set where the Active Directory driver is stored, then click *Search*.

3 Click the upper right corner of the Active Directory driver, then click *Edit Properties*.



4 In the *Driver Configuration* tab, click *Edit XML* under *Driver Parameters*.


Identity Manager
Server Variables
General
Driver Configuration | Global Config Values | Named Passwords | Engine Control Values | Log Level | Driver Image | Security Equals | Filter | Edit Filter XML | Misc | Excluded Objects | Driver Manifest | Driver Inspector | Driver Cache Inspector | Inspector

○ Disabled

Driver Parameters

IDMTEST.novell

Edit XML

Driver Settings

Polling Interval (min.)	1
Authentication Method	Negotiate
Use Signing (yes/no)	no
Use Sealing (yes/no)	no
Use SSL (yes/no)	no
Heart Beat	0
Password Sync Timeout (minutes):	5
Use CDOEXM for Exchange (yes/no)	yes

5 Click the *Enable XML editing* check box.

6 Search for the heading `<header display-name=“Exchange Options”/>`.

7 Change the following lines:

Old XML	New XML
`<definition display-name="Use CDOEXM for Exchange (yes/no)" name="use-CDOEXM" type="enum">`	`<definition display-name="Exchange Management interface type (use-cdoexm/use-post-cdoexm)" name="exch-api-type" type="enum">`
`<enum-choice display-name="Yes">yes</enum-choice>`	`<enum-choice display-name="use-cdoexm">use-cdoexm</enum-choice>`
`<enum-choice display-name="No">no</enum-choice>`	`<enum-choice display-name="use-post-cdoexm">use-post-cdoexm</enum-choice>`
`<definition display-name="Allow CDOEXM Exchange mailbox move (yes/no)" name="cdoexm-move" type="enum">`	`<definition display-name="Allow Exchange mailbox move (yes/no)" name="exch-move" type="enum">`
`<definition display-name="Allow CDOEXM Exchange mailbox delete (yes/no)" name="cdoexm-delete" type="enum">`	`<definition display-name="Allow Exchange mailbox delete (yes/no)" name="exch-delete" type="enum">`

8 Click *OK* twice to save the changes.

DirXML Command Line Utility

D

The DirXML® Command Line utility allows you to use a command line interface to manage the driver. You can create scripts to manage the driver with the commands.

The utility and scripts are installed on all platforms during the Identity Manager installation. The utility is installed to the following locations:

- Windows: `\Novell\Nds\dxcmd.bat`
- NetWare®: `sys:\system\dxcmd.ncf`
- UNIX: `/usr/bin/dxcmd`

There are two different methods for using the DirXML Command Line utility:

- Section D.1, "Interactive Mode," on page 153
- Section D.2, "Command Line Mode," on page 162

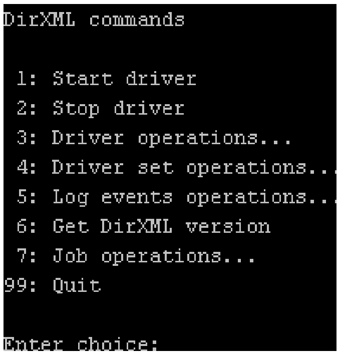
D.1 Interactive Mode

The interactive mode provides a text interface to control and use the DirXML Command Line utility.

1 At the console, enter `dxcmd`.

2 Enter the name of a user with sufficient rights to the Identity Manager objects, such as admin.novell.

3 Enter the user's password.



```
DirXML commands

 1: Start driver
 2: Stop driver
 3: Driver operations...
 4: Driver set operations...
 5: Log events operations...
 6: Get DirXML version
 7: Job operations...
99: Quit

Enter choice:
```

4 Enter the number of the command you want to perform.

Table D-1 on page 154 contains the list of options and what functionality is available.

5 Enter 99 to quit the utility.

NOTE: If you are running eDirectory™ 8.8 on UNIX or Linux, you must specify the -host and -port parameters. For example, `dxcmd -host 10.0.0.1 -port 524`. If the parameters are not specified, a jclient error occurs:

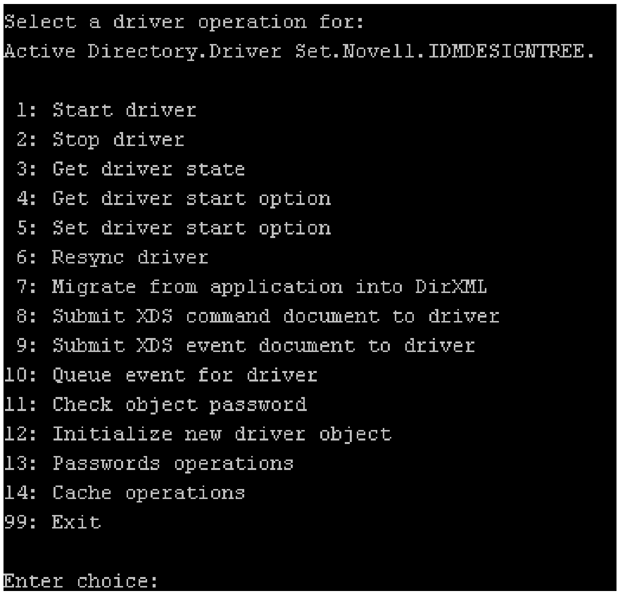
```
novell.jclient.JCException: connect (to address) 111 UNKNOWN ERROR
```

By default, eDirectory 8.8 is not listening to localhost. The DirXML Command Line utility needs to resolve the server IP address or hostname and the port to be able to authenticate.

Table D-1 *Interactive Mode Options*

Option	Description
1: *Start Driver*	Starts the driver. If there is more than one driver, each driver is listed with a number. Enter the number of the driver to start the driver.
2: *Stop Driver*	Stops the driver. If there is more than one driver, each driver is listed with a number. Enter the number of the driver to stop the driver.
3: *Driver operations*	Lists the operations available for the driver. If there is more than one driver, each driver is listed with a number. Enter the number of the driver to see the operations available. See Table D-2 on page 155 for a list of operations.
4: *Driver set operations*	Lists the operations available for the driver set. ◆ 1: Associate driver set with server ◆ 2: Disassociate driver set from server ◆ 99: Exit
5: *Log events operations*	Lists the operations available for logging events through Novell® Audit. See Table D-5 on page 159 for a description of these options.
6: *Get DirXML version*	Lists the version of the Identity Manager installed.
7: *Job operations*	Manages jobs created for Identity Manager.
99: *Quit*	Exits the DirXML Command Line utility

Figure D-1 *Driver Options*



```
Select a driver operation for:
Active Directory.Driver Set.Novell.IDMDESIGNTREE.

 1: Start driver
 2: Stop driver
 3: Get driver state
 4: Get driver start option
 5: Set driver start option
 6: Resync driver
 7: Migrate from application into DirXML
 8: Submit XDS command document to driver
 9: Submit XDS event document to driver
10: Queue event for driver
11: Check object password
12: Initialize new driver object
13: Passwords operations
14: Cache operations
99: Exit

Enter choice:
```

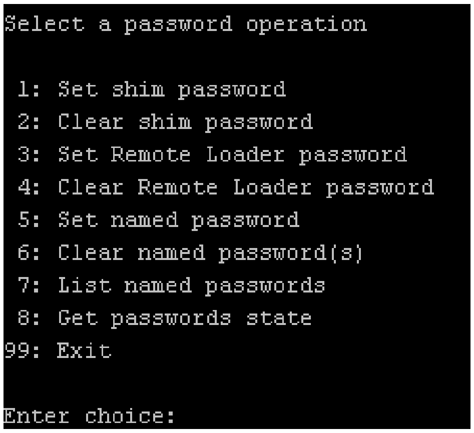
Table D-2 *Driver Options*

Options	Description
1: *Start driver*	Starts the driver.
2: *Stop driver*	Stops the driver.
3: *Get driver state*	Lists the state of the driver. ◆ 0 - Driver is stopped ◆ 1 - Driver is starting ◆ 2 - Driver is running ◆ 3 - Driver is stopping
4: *Get driver start option*	Lists the current driver start option. ◆ 1 - Disabled ◆ 2 - Manual ◆ 3 - Auto
5: *Set driver start option*	Changes the start option of the driver. ◆ 1 - Disabled ◆ 2 - Manual ◆ 3 - Auto ◆ 99 - Exit
6: *Resync driver*	Forces a resynchronization of the driver. It prompts for a time delay: *Do you want to specify a minimum time for resync? (yes/no).* If you enter Yes, specify the date and time you want the resynchronization to occur: *Enter a date/time (format 9/27/05 3:27 PM)*. If you enter No, the resynchronization occurs immediately.
7: *Migrate from application into DirXML*	Processes an XML document that contains a query command: *Enter filename of XDS query document:* Create the XML document that contains a query command by using the Novell `nds.dtd` (http://developer.novell.com/ndk/doc/dirxml/dirxmlbk/ref/ndsdtd/query.html). Examples: NetWare: `sys:\files\query.xml` Windows: `c:\files\query.xml` Linux: `/files/query.xml`

Options	Description
8: *Submit XDS command document to driver*	Processes an XDS command document: *Enter filename of XDS command document:* Examples: NetWare: `sys:\files\user.xml` Windows: `c:\files\user.xml` Linux: `/files/user.xml` *Enter name of file for response:* Examples: NetWare: `sys:\files\user.log` Windows: `c:\files\user.log` Linux: `/files/user.log`
9: Submit XDS event document to driver	Processes an XDS event document: *Enter filename of XDS event document:* Examples: NetWare: `sys:\files\add.xml` Windows: `c:\files\add.xml` Linux: `/files/add.xml`
10: Queue event for driver	Adds an event to the driver queue: *Enter filename of XDS event document:* Examples: NetWare: `sys:\files\add.xml` Windows: `c:\files\add.xml` Linux: `/files/add.xml`
11: *Check object password*	Validates that an object's password in the connected system is associated with a driver. It matches the object's eDirectory password (Distribution Password, used with Universal Password). *Enter user name*:
12: *Initialize new driver object*	Performs an internal initialization of data on a new Driver object. This is only for testing purposes.
13: *Password operations*	There are nine Password options. See Table D-3 on page 157 for a description of these options.
14: *Cache operations*	There are five Cache operations. See Table D-4 on page 158 for a description of these options.

Options	Description
99: *Exit*	Exits the driver options.

Figure D-2 *Password Operations*



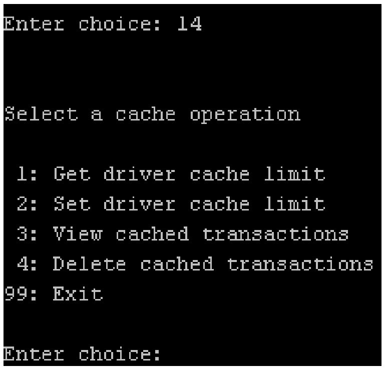
```
Select a password operation

 1: Set shim password
 2: Clear shim password
 3: Set Remote Loader password
 4: Clear Remote Loader password
 5: Set named password
 6: Clear named password(s)
 7: List named passwords
 8: Get passwords state
99: Exit

Enter choice:
```

Table D-3 *Password Operations*

Operation	Description
1: *Set shim password*	Sets the application password. This is the password of the user account you are using to authenticate into the connected system with.
2: *Clear shim password*	Clears the application password.
3: *Set Remote Loader password*	The Remote Loader password is used to control access to the Remote Loader instance. Enter the Remote Loader password, then confirm the password by typing it again.
4: *Clear Remote Loader password*	Clears the Remote Loader password so no Remote Loader password is set on the Driver object.
5: *Set named password*	Allows you to store a password or other pieces of security information on the driver. See Section 8.9, “Storing Driver Passwords Securely with Named Passwords,” on page 98 for more information. There are four prompts to fill in: ◆ *Enter password name:* ◆ *Enter password description:* ◆ *Enter password:* ◆ *Confirm password:*

Operation	Description
6: *Clear named passwords*	Clears a specified named password or all named passwords that are stored on the driver object. *Do you want to clear all named passwords? (yes/ no).* If you enter Yes, all Named Passwords are cleared. If you enter No, you are prompted to specify the password name that you want to clear.
7: *List named passwords*	Lists all named passwords that are stored on the driver object. It lists the password name and the password description.
8: *Get password state*	Lists if a password is set for: ◆ Driver Object password ◆ Application password ◆ Remote loader password The dxcmd utility allows you to set the Application password and the Remote Loader password. You cannot set the Driver Object password with this utility. It shows if the password has been set or not.
99: *Exit*	Exits the current menu and takes you back to the Driver options.

Figure D-3 *Cache Operations*



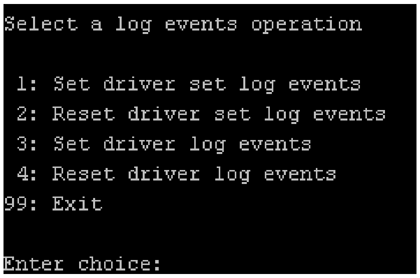
```
Enter choice: 14


Select a cache operation

 1: Get driver cache limit
 2: Set driver cache limit
 3: View cached transactions
 4: Delete cached transactions
99: Exit

Enter choice:
```

Table D-4 *Cache Operations*

Operation	Description
1: *Get driver cache limit*	Displays the current cache limit that is set for the driver.
2: *Set driver cache limit*	Sets the driver cache limit in kilobytes. A value of 0 is unlimited.

Operation	Description
3: *View cached transactions*	A text file is created with the events that are stored in cache. You can select the number of transactions to view. ◆ *Enter option token* (default=0): ◆ *Enter maximum transactions records to return* (default=1): ◆ *Enter name of file for response:*
4: *Delete cached transactions*	Deletes the transactions stored in cache. ◆ *Enter position token* (default=0): ◆ *Enter event-id value of first transaction record* to delete (optional): ◆ *Enter number of transaction records to delete* (default=1):
99: *Exit*	Exits the current menu and takes you back to the Driver options.

Figure D-4 *Log Event Operations*



```
Select a log events operation

 1: Set driver set log events
 2: Reset driver set log events
 3: Set driver log events
 4: Reset driver log events
99: Exit

Enter choice:
```

Table D-5 *Log Events Operations*

Operation	Description
1: *Set driver set log events*	Allows you to log driver set events through Novell Audit. There are 49 items you can select to log. See Table D-6 on page 160 for a list of these options. Type the number of the item you want to log. After the items are selected, enter 99 to accept the selections.
2: *Reset driver set log events*	Resets all of the log event options.
3: *Set driver log events*	Allows you to log driver events through Novell Audit. There are 49 items to select to log. See Table D-6 on page 160 for a list of these options. Type the number of the item you want to log. After the items are selected, enter 99 to accept the selections.

Operation	Description
4: *Reset driver log events*	Resets all of the log event options.
99: *Exit*	Exits the log events operations menu.

Table D-6 *Driver Set and Driver Log Events*

Options
1: Status success
2: Status retry
3: Status warning
4: Status error
5: Status fatal
6: Status other
7: Query elements
8: Add elements
9: Remove elements
10: Modify elements
11: Rename elements
12: Move elements
13: Add-association elements
14: Remove-association elements
15: Query-schema elements
16: Check-password elements
17: Check-object-password elements
18: Modify-password elements
19: Sync elements
20: Pre-transformed XDS document from shim
21: Post input transformation XDS document
22: Post output transformation XDS document
23: Post event transformation XDS document
24: Post placement transformation XDS document
25: Post create transformation XDS document
26: Post mapping transformation <inbound> XDS document
27: Post mapping transformation <outbound> XDS document

Options
28: Post matching transformation XDS document
29: Post command transformation XDS document
30: Post-filtered XDS document <Publisher>
31: User agent XDS command document
32: Driver resync request
33: Driver migrate from application
34: Driver start
35: Driver stop
36: Password sync
37: Password request
38: Engine error
39: Engine warning
40: Add attribute
41: Clear attribute
42: Add value
43: Remove value
44: Merge entire
45: Get named password
46: Reset Attributes
47: Add Value - Add Entry
48: Set SSO Credential
49: Clear SSO Credential
50: Set SSO Passphrase
51: User defined IDs
99: Accept checked items

Table D-7 *Enter Table Title Here*

Options	Description
1: *Get available job definitions*	Allows you to select an existing job. *Enter the job number:* *Do you want to filter the job definitions by containment?* Enter Yes or No *Enter name of the file for response:* Examples: NetWare: `sys:\files\user.log` Windows: `c:\files\user.log` Linux: `/files/user.log`
2: *Operations on specific job object*	Allows you to perform operations for a specific job.

D.2 Command Line Mode

The command line mode allows you to use script or batch files. Table D-8 on page 162 lists the different options that are available.

To use the command line options, decide which items you want to use and string them together.

Example: `dxcmd -user admin.headquarters -host 10.0.0.1 -password n0vell -start test.driverset.headquarters`

This example command starts the driver.

Table D-8 *Command Line Options*

Option	Description
Configuration	
-user <*user name*>	Specify the name of a user with administrative rights to the drivers you want to test.
-host <*name or IP address*>	Specify the IP address of the server where the driver is installed.
-password <*user password*>	Specify the password of the user specified above.
-port <*port number*>	Specify a port number, if the default port is not used.
-q <*quiet mode*>	Displays very little information when a command is executed.
-v <*verbose mode*>	Displays detailed information when a command is executed.

Option	Description
-s *<stdout>*	Writes the results of the `dxcmd` command to `stdout`.
-? *<show this message>*	Displays the help menu.
-help *<show this message>*	Displays the help menu.
Actions	
-start *<driver dn>*	Starts the driver.
-stop *<driver dn>*	Stops the driver.
-getstate *<driver dn>*	Shows the state of the driver as running or stopped.
-getstartoption *<driver dn>*	Shows the startup option of the driver.
-setstartoption *<driver dn> <disabled|manual|auto> <resync|noresync>*	Sets how the driver starts if the server is rebooted. Sets whether the objects are to be resynchronized when the driver restarts.
-getcachelimit *<driver dn>*	Lists the cache limit set for the driver.
-setcachelimit *<driver dn> <0 or positive integer>*	Sets the cache limit for the driver.
-migrateapp *<driver dn> <filename>*	Processes an XML document that contains a query command. Create the XML document that contains a query command by using the Novell `nds.dtd` (http://www.novell.com/documentation/idm35/index.html?page=/documentation/idm35/policy_dtd/data/dtdndsoverview.html#dtdndsoverview).
-setshimpassword *<driver dn> <password>*	Sets the application password. This is the password of the user account you are using to authenticate into the connected system with.
-clearshimpassword *<driver dn> <password>*	Clears the application password.
-setremoteloaderpassword *<driver dn> <password>*	Sets the Remote Loader password. The Remote Loader password is used to control access to the Remote Loader instance.
<clearremoteloaderpassword *<driver dn>*	Clears the Remote Loader password.

Option	Description
-sendcommand *<driver dn> <input filename> <output filename>*	Processes an XDS command document. Specify the XDS command document as the input file. Examples: NetWare: `sys:\files\user.xml` Windows: `c:\files\user.xml` Linux: `/files/user.log` Specify the output filename to see the results. Examples: NetWare: `sys:\files\user.log` Windows: `c:\files\user.log` Linux: `/files/user.log`
-sendevent *<driver dn> <input filename>*	Submits a document to the driver's Subscriber channel, bypassing the driver cache. The document is processed ahead of anything that might be in the cache at the time of the submission. It also means that the submission fails if the driver is not running.
-queueevent *<driver dn> <input filename>*	Submits a document to the driver's Subscriber channel by queuing the document in the driver cache. The document gets processed after anything that might be in the cache at the time of the submission. The submission won't fail if the driver isn't running.
-setlogevents *<dn> <integer ...>*	Sets Novell Audit log events on the driver. The integer is the option of the item to log. See Table D-6 on page 160 for the list of the integers to enter.
-clearlogevents *<dn>*	Clears all Novell Audit log events that are set on the driver.
-setdriverset *<driver set dn>*	Associates a driver set with the server.
-cleardriverset	Clears the driver set association from the server.
-getversion	Shows the version of Identity Manager that is installed.
-initdriver object *<dn>*	Performs an internal initialization of data on a new Driver object. This is only for testing purposes.
-setnamedpassword *<driver dn> <name> <password> [description]*	Sets named passwords on the driver object. You specify the name, the password, and the description of the named password.
-clearnamedpassword *<driver dn> <name>*	Clears a specified named password.
-startjob *<job dn>*	Starts the specified job.

Option	Description
-abortjob *<job dn>*	Aborts the specified job.
-getjobrunningstate *<job dn>*	Returns the specified job's running state.
-getjobenabledstate *<job dn>*	Returns the specified job's enabled state.
-getjobnextruntime *<job dn>*	Returns the specified job's next run time.
-updatejob *<job dn>*	Updates the specified job.
-clearallnamedpaswords *<driver dn>*	Clears all named passwords set on a specific driver.

If a command line is executed successfully, it returns a zero. If the command line returns anything other than zero, it is an error. For example 0 means success, and -641 means invalid operation. -641 is an eDirectory error code. Table D-9 on page 165 contains other values for specific command line options.

Table D-9 *Command Line Option Values*

Command Line Option	Values
-getstate	0- stopped
	1- starting
	2- running
	3- shutting down
	11- get schema
	Anything else that is returned is an error.
-getstartoption	0- disabled
	1- manual
	2- auto
	Anything else that is returned is an error.
-getcachelimit	0- unlimited
	Anything else that is returned is an error.
-getjobrunningstate	0- stopped
	1- running
	Anything else that is returned is an error.
-getjobenabledstate	0- disabled
	1- enabled
	2- configuration error
	Anything else that is returned is an error.

Command Line Option	Values
-getjobnextruntime	Return is the next scheduled time for the job in eDirectory time format (number of seconds since 00:00:00 Jan 1, 1970 UTC).

Properties of the Driver

E

There are many different fields and values for the driver. Sometimes the information is displayed differently in iManager than in Designer. This section is a reference for all of the fields on the driver as displayed in iManager and Designer.

The information is presented from the viewpoint of iManager. If a field is different in Designer, it is marked with an icon.

E.1 Driver Configuration

In iManager:

1. Click *Identity Manager* > *Identity Manager Overview*, then click *Search* to search for the driver set that is associated with the driver.
2. Browse to the driver, then click the upper right corner of the driver icon.
3. Click *Edit Properties* > *Driver Configuration.*

In Designer:

1. Open a project in the Modeler, then right-click the driver line and click *Properties* > *Driver Configuration.*

There are different sections under *Driver Configuration.* In this document, each section is listed in a table. The table contains a description of the fields, and the default value or an example of the value that should be specified in the field.

E.1.1 Driver Module

The driver module changes the driver from running locally to running remotely or the reverse.

In iManager:

1. Click *Identity Manager > Identity Manager Overview*, then click *Search* to search for the driver set that is associated with the driver.
2. Browse to the driver, then click the upper right corner of the driver icon.
3. Click *Edit Properties > Driver Configuration > Driver Module.*

   See Table E-1 for a list of the driver module options.

In Designer:

1. Open a project in the Modeler, then right-click the driver line and select *Properties > Driver Configuration.*
2. Select the *Driver Module* tab.

   See Table E-1 for a list of the driver module options.

Table E-1 *Driver Module Options*

Option	**Description**
Java	Used to specify the name of the Java class that is instantiated for the shim component of the driver. This class can be located in the `classes` directory as a class file, or in the `lib` directory as a `.jar` file. If this option is selected, the driver is running locally.
Native	Used to specify the name of the `.dll` file that is instantiated for the application shim component of the driver. If this option is selected, the driver is running locally.
Connect to Remote Loader	Used when the driver is connecting remotely to the connected system.
Remote Loader Client Configuration for Documentation	Includes the Remote Loader client configuration information in the driver documentation that is generated by Designer.

E.1.2 Driver Object Password

In iManager:

1. Click *Identity Manager > Identity Manager Overview*, then click *Search* to search for the driver set that is associated with the driver.
2. Browse to the driver, then click the upper right corner of the driver icon.
3. Click *Edit Properties > Driver Configuration > Driver Object Password > Set Password*.

   See Table E-2 for more information.

In Designer:

1. Open a project in the Modeler, then right-click the driver line and click *Properties > Driver Configuration*.
2. Click *Driver Module > Connect to Remote Loader > Driver Object Password > Set Password*.

   See Table E-2 for more information.

Table E-2 *Driver Object Password*

Option	Description
Driver Object Password	Use this option to set a password for the driver object. If you are using the Remote Loader, you must enter a password on this page or the remote driver does not run. This password is used by the Remote Loader to authenticate itself to the remote driver shim.

E.1.3 Authentication

The authentication section stores the information required to authenticate to the connected system.

In iManager:

1. Click *Identity Manager > Identity Manager Overview*, then click *Search* to search for the driver set that is associated with the driver.
2. Browse to the driver, then click the upper right corner of the driver icon.
3. Click *Edit Properties > Driver Configuration > Authentication*.

   See Table E-3 for a list of the driver authentication parameters.

In Designer:

1. Open a project in the Modeler, then right-click the driver line and select *Properties > Driver Configuration*.
2. Click *Authentication*.

   See Table E-3 for a list of the driver authentication parameters.

Table E-3 *Authentication Parameters*

Option	Description
Authentication ID or *User ID*	Specify a user application ID. This ID is used to pass Identity Vault subscription information to the application. Example: `Administrator`
Authentication Context or *Connection Information*	Specify the IP address or name of the server the application shim should communicate with.

Option	Description
Remote Loader Connection Parameters or *Host name* Port *KMO* *Other parameters*	Used only if the driver is connecting to the application through the Remote Loader. The parameter to enter is `hostname=xxx.xxx.xxx.xxx port=xxxx kmo=certificatename`, when the host name is the IP address of the application server running the Remote Loader server and the port is the port the Remote Loader is listening on. The default port for the Remote Loader is 8090. The `kmo` entry is optional. It is only used when there is an SSL connection between the Remote Loader and the Metadirectory engine. Example: `hostname=10.0.0.1 port=8090 kmo=IDMCertificate`
Driver Cache Limit (kilobytes) or *Cache limit (KB)*	Specify the maximum event cache file size (in KB). If it is set to zero, the file size is unlimited. Click *Unlimited* to set the file size to unlimited in Designer.
Application Password or *Set Password*	Specify the password for the user object listed in the *Authentication ID* field.
Remote Loader Password or *Set Password*	Used only if the driver is connecting to the application through the Remote Loader. The password is used to control access to the Remote Loader instance. It must be the same password specified during the configuration of the Remote Loader on the connected system.

E.1.4 Startup Option

The Startup Option allows you to set the driver state when the Identity Manager server is started.

In iManager:

1. Click *Identity Manager > Identity Manager Overview*, then click *Search* to search for the driver set that is associated with the driver.
2. Browse to the driver, then click the upper right corner of the driver icon.
3. Click *Edit Properties > Driver Configuration > Startup Option*.

   See Table E-4 for a list of the startup options.

In Designer:

1. Open a project in the Modeler, then right-click the driver line and select *Properties > Driver Configuration*.
2. Click *Startup Option*.

   See Table E-4 for a list of the startup options.

Table E-4 *Startup Options*

Option	Description
Auto start	The driver starts every time the Identity Manager server is started.
Manual	The driver does not start when the Identity Manager server is started. The driver must be started through Designer or iManager.
Disabled	The driver has a cache file that stores all of the events. When the driver is set to Disabled, this file is deleted and no new events are stored in the file until the driver state is changed to Manual or Auto Start.
Do not automatically synchronize the driver	This option only applies if the driver is deployed and was previously disabled. If this is not selected, the driver re-synchronizes the next time it is started.

E.1.5 Driver Parameters

In iManager:

1. Click *Identity Manager > Identity Manager Overview*, then click *Search* to search for the driver set that is associated with the driver.
2. Browse to the driver, then click the upper right corner of the driver icon.
3. Click *Edit Properties > Driver Configuration > Driver Parameters*.

   See Table E-5 for a list of the driver parameters.

In Designer:

1. Open a project in the modeler, then right-click the driver line and select *Properties > Driver Configuration*.
2. Click *Driver Parameters*.

   See Table E-5 for a list of the driver parameters.

Table E-5 *Driver Parameters*

Parameter	Description
Driver Settings > Authentication Options	
Show authentication options	The options are *show* or *hide*. It enables you to see and change the authentication options for the driver.
Authentication Method	The method of authentication to Active Directory. *Negotiate* uses Microsoft's security package to negotiate the logon type. Typically Kerberos or NTLM is selected. *Simple* uses LDAP style simple bind for logon. If you want to use Password Synchronization, select *Negotiate*.

Parameter	Description
Digitally sign communications	Select *Yes* to digitally sign communication between the driver shim and Active Directory. This does not hide the data from view on the network, but it reduces the change of security attacks. Signing only works when you use the *Negotiate* authentication method and the underlying security provider selects NTLM2 or Kerberos for its protocol. Do not use this option with SSL. Select *No* to have communications not signed.
Digitally sign and seal communications	Select *Yes* to digitally encrypt communication between the driver shim and the Active Directory database. Sealing only works when you the *Negotiate* authentication method and the underlying security provider selects NTLM2 or Kerberos for its protocols. Do not use this options with SSL. Select *No* to not have communication between the driver shim and the Active Directory database signed and sealed.
Use SSL for encryption	Select *Yes* to digitally encrypt communication between the driver shim and the Active Directory database. This option can be used with the *Negotiate* or *Simple* authentication methods. SSL requires that the Microsoft server running the driver shim imports the domain controller's server certificate imported. For more information, see Securing Windows 200 Server (http://www.microsoft.com/technet/security/prodtech/windows2000/secwin2k/swin2kad.mspx).
Logon and impersonate	Select *Yes* to logon and impersonate the driver authentication account for CDOEXM (Collaboration Data Object for Exchange Management) and Password Set support. The driver performs a local logon. The authentication account must have the proper rights assignment. For more information, see Section 2.4, "Creating an Administrative Account," on page 28. If *No* is selected, the driver performs a network logon only.
Driver Settings > Exchange Options	
Show Microsoft Exchange options	Select *show* to display the Microsoft Exchange options. These parameters control whether the driver shim uses the Microsoft CDOEXM Exchange management APIs and whether to interpret changes in the homeMDB attribute as a Move or a Delete of the mailbox. Select *hide* if you are not synchronizing Exchange accounts.
Use CDOEXM for Exchange (yes/no)	Select *Yes* to enable the driver shim to intercept changes to the Active Directory homeMDB attribute and calls into the CDOEXM subsystem. The value selected here is stored in the driver shim configuration. Select *No* if you are not synchronizing Exchange accounts.

Parameter	Description
Allow CDOEXM Exchange mailbox move (yes/no)	Select *Yes* to enable the driver shim to intercept modifications to the Active Directory homeMDB attribute and calls into the CDOEXM subsystem to move the mailboxes to the new message data store. Select *No* if you do not want mailboxes moved when the Active Directory account is moved.
Allow CDOEXM Exchange mailbox delete (yes/no)	Select *Yes* to enable the driver shim to intercept removals of the Active Directory homeMDB attribute and calls into the CDOEXM subsystem to delete the mailbox. Select *No* if you don't want to delete the mailbox account when the Active Directory account is deleted.
Driver Settings > Access Options	
Show access options	Select *show* to display the domain controller access options. These parameters control the scope of the Active Directory queries along with several Publisher polling and timeout parameters. Select *hide* to hide the domain controller access options.
Driver Polling Interval	Specify the number of minutes to delay before querying the Active Directory data base for changes. A larger number reduces the load on the Active Directory database, but it also reduces the responsiveness of the driver. The default value is 1 minute.
Publisher heartbeat interval	Allows the driver to send a periodic status message on the Publisher channel when there has been no Publisher channel traffic for the given number of seconds. The default value is 0 seconds, which means this is not enabled.
Password Sync Timeout (minutes)	Specify the number of minutes for the driver to attempt to synchronize a given password. The driver does not try to synchronize the password after this interval has been exceeded. It is recommended that this value be set to at least three times the value of the polling interval. For example, if the *Driver Polling Interval* is set to 10 minutes, set the *Password Sync Timeout* to 30 minutes. If this value is set to 0, then password synchronization is disabled for this driver. The default value is 5 minutes.
Search domain scope	The driver shim reads information from other domains when objects in those domains are referenced. If the account you use for authentication has no rights in the other domain, the reads might fail. Select *Yes* to enable this option if you get access errors during regular operations. By default, it is set to *No*.

E.2 Global Configuration Values

Global configuration values (GCVs) allow you to specify settings for the Identity Manager features such as password synchronization and driver heartbeat, as well as settings that are specific to the function of an individual driver configuration. Some GCVs are provided with the drivers, but you can also add your own.

IMPORTANT: Password synchronization settings are GCVs, but it's best to edit them in the graphical interface provided on the Server Variables page for the driver, instead of the GCV page. The Server Variables page that shows Password Synchronization settings is accessible as a tab like other driver parameters, or by clicking *Password Management > Password Synchronization*, searching for the driver, and clicking the driver name. The page contains online help for each Password Synchronization setting.

In iManager:

1. Click *Identity Manager > Identity Manager Overview*, then click *Search* to search for the driver set that is associated with the driver.
2. Browse to the driver, then click the upper right corner of the driver icon.
3. Click *Edit Properties > Global Config Values*.

In Designer:

1. Open a project in the Modeler, then right-click the driver line and select *Properties > Global Config Values.*

There are different categories of global configuration values.

- Table E-6, "Global Configuration Values > Driver parameters," on page 174
- Table E-7, "Global Configuration Values > Entitlement selections," on page 174
- Table E-8, "Global Configuration Values > Password Management," on page 175

Table E-6 *Global Configuration Values > Driver parameters*

Option	Description
Connected System or Driver Name	Contains the name of the connected system, application, or Identity Manager driver. This value is used by e-mail notification templates to identify the source of the notification messages. For more information, see "Configuring E-Mail Notification" in the *Novell Identity Manager 3.5.1 Administration Guide*.

Table E-7 *Global Configuration Values > Entitlement selections*

Option	Description
When account entitlement revoked	Select the desired action in the Active Directory data base when a User Account Entitlement is revoked. The options are *Disable Account* or *Delete Account*.

For *Password Configuration*, you should only edit the first two settings listed here. The others are GCVs regarding Password Synchronization that are common to all drivers. They should be edited using iManager in *Passwords* > *Password Synchronization*, not here. Some of them have dependencies on each other that are represented only in the iManager interface. They are explained in "Password Synchronization across Connected Systems" in the *Novell Identity Manager 3.5.1 Administration Guide*.

Table E-8 *Global Configuration Values > Password Management*

Option	Description
Show password management policy	Select *show* to display the global configuration values for password management. Select *hide* to not have the password management global configuration values displayed.
Application accepts passwords from Identity Manager	If *True*, allows passwords to flow from the Identity Manager data store to the connected system.
Identity Manager accepts passwords from application	If *True*, allows passwords to flow from the connected system to Identity Manager.
Publish passwords to NDS password	Use the password from the connected system to set the non-reversible NDS® password in eDirectory.
Publish passwords to Distribution Password	Use the password from the connected system to set the NMAS™ Distribution Password used for Identity Manager password synchronization.
Require password policy validation before publishing passwords	If *True*, applies NMAS password policies during publish password operations. The password is not written to the data store if it does not comply.
Reset user's external system password to the Identity Manager password on failure	If *True*, on a publish Distribution Password failure, attempt to reset the password in the connected system using the Distribution Password from the Identity Manager data store.
Notify the user of password synchronization failure via e-mail	If *True*, notify the user by e-mail of any password synchronization failures.

E.3 Named Passwords

Identity Manager allows you to store multiple passwords securely for a particular driver. This functionality is referred to as Named Passwords. Each different password is accessed by a key, or name.

You can also use the Named Passwords feature to store other pieces of information securely, such as a user name. To configured Named Passwords, see Section 8.9, "Storing Driver Passwords Securely with Named Passwords," on page 98.

In iManager:

1 Click *Identity Manager* > *Identity Manager Overview*, then click *Search* to search for the driver set that is associated with the driver.

2 Browse to the driver, then click the upper right corner of the driver icon.

3 Click *Edit Properties* > *Named Passwords*.

In Designer:

1 Open a project in the Modeler, then right-click the driver line and select *Properties* > *Named Passwords.*

E.4 Engine Control Values

The engine control values are a means through which certain default behaviors of the Metadirectory engine can be changed. The values can only be accessed if a server is associated with the Driver Set object.

In iManager:

1 Click *Identity Manager* > *Identity Manager Overview*, then click *Search* to search for the driver set that is associated with the driver.

2 Browse to the driver, then click the upper right corner of the driver icon.

3 Click *Edit Properties* > *Engine Control Values*.

See Table E-9 for a list of the engine control values.

In Designer:

1 In the Modeler, right-click the driver line.

2 Select *Properties* > *Engine Control Values*.

3 Click the tooltip icon to the right of the *Engine Controls For Server* field. If a server is associated with the Identity Vault, and if you are authenticated, the engine control values display in the large pane.

See Table E-9 for a list of the engine control values.

Table E-9 *Engine Control Values*

Option	Description
Subscriber channel retry interval in seconds	The Subscriber channel retry interval controls how frequently the Metadirectory engine retries the processing of a cached transaction after the application shim's Subscriber object returns a retry status.
Qualified form for DN-syntax attribute values	The qualified specification for DN-syntax attribute values controls whether values for DN-syntax attribute values are presented in unqualified slash form or qualified slash form. A True setting means the values are presented in qualified form.
Qualified form from rename events	The qualified form for rename events controls whether the new-name portion of rename events coming from the Identity Vault is presented to the Subscriber channel with type qualifiers. For example, CN=. A True setting means the names are presented in qualified form.

Option	Description
Maximum eDirectory replication wait time in seconds	The maximum eDirectory™ replication wait time controls the maximum time that the Metadirectory engine waits for a particular change to replicate between the local replica and a remote replica. This only affects operations where the Metadirectory engine is required to contact a remote eDirectory server in the same tree to perform an operation and might need to wait until some change has replicated to or from the remote server before the operation can be completed (for example, object moves when the Identity Manager server does not hold the master replica of the moved object; file system rights operations for Users created from a template.)
Use non-compliant backwards-compatible mode for XSLT	This control sets the XSLT processor used by the Metadirectory engine to a backward-compatible mode. The backward-compatible mode causes the XSLT processor to use one or more behaviors that are not XPath 1.0 and XSLT 1.0 standards-compliant. This is done in the interest of backward-compatibility with existing DirXML® style sheets that depend on the non-standard behaviors. For example, the behavior of the XPath “!=” operator when one operand is a node set and the other operand is other than a node set is incorrect in DirXML releases up to and including Identity Manager 2.0. This behavior has been corrected; however, the corrected behavior is disabled by default through this control in favor of backward-compatibility with existing DirXML style sheets.
Maximum application objects to migrate at once	This control is used to limit the number of application objects that the Metadirectory engine requests from an application during a single query that is performed as part of a Migrate Objects from Application operation. If java.lang.OutOfMemoryError errors are encountered during a Migrate from Application operation, this number should be set lower than the default. The default is 50. **NOTE:** This control does not limit the number of application objects that can be migrated; it merely limits the batch size.
Set creatorsName on objects created in Identity Vault	This control is used by the Identity Manager engine to determine if the creatorsName attribute should be set to the DN of this driver on all objects created in the Identity Vault by this driver. Setting the creatorsName attribute allows for easily identifying objects created by this driver, but also carries a performance penalty. If not set, the creatorsName attribute defaults to the DN of the NCP™ Server object that is hosting the driver.
Write pending associations	This control determines whether the Identity Manager engine writes a pending association on an object during Subscriber channel processing. Writing a pending association confers little or no benefit but does incur a performance penalty. Nevertheless, the option exists to turn it on for backward compatibility.

Option	Description
Use password event values	This control determines the source of the value reported for the nspmDistributionPassword attribute for Subscriber channel Add and Modify events. Setting the control to False means that the current value of the nspmDistributionPassword is obtained and reported as the value of the attribute event. This means that only the current password value is available. This is the default behavior. Setting the control to True means that the value recorded with the eDirectory event is decrypted and is reported as the value of the attribute event. This means that both the old password value (if it exists) and the replacement password value at the time of the event are available. This is useful for synchronizing passwords to certain applications that require the old password to enable setting a new password.
Enable password synchronization status reporting	This control determines whether the Identity Manager engine reports the status of Subscriber channel password change events. Reporting the status of Subscriber channel password change events allows applications such as the Identity Manager User Application to monitor the synchronization progress of a password change that should be synchronized to the connected application.

E.5 Log Level

Every driver set and driver has a log level field where you can define the level of errors that should be tracked. The level you indicate here determines which messages are available to the logs. By default, the log level is set to track error messages (this also includes fatal messages). Change the log level if you want to track additional message types.

Novell® recommends that you use Novell Audit instead of setting the log levels. See "Integrating Identity Manager with Novell Audit" in the *Identity Manager 3.5.1 Logging and Reporting*.

In iManager:

1 Click *Identity Manager* > *Identity Manager Overview*, then click *Search* to search for the driver set that is associated with the driver.

2 Browse to the driver, then click the upper right corner of the driver icon.

3 Click *Edit Properties* > *Log Level*.

   See Table E-10 for a list of the driver log levels.

In Designer:

1 Open a project in the Modeler, then right-click the driver line and select *Properties* > *Driver Log Level*.

   See Table E-10 for a list of the driver log levels.

Table E-10 Driver Log Levels

Option	Description
Use log settings from the DriverSet	If this is selected, the driver logs events as the options are set on the Driver Set object.
Log errors	Logs just errors
Log errors and warnings	Logs errors and warnings
Log specific events	Logs the events that are selected. Click the icon to see a list of the events.
Only update the last log time	Updates the last log time.
Logging off	Turns logging off for the driver.
Turn off logging to DriverSet, Subscriber and Publisher logs	If selected, turns all logging off for this driver on the Driver Set object, Subscriber channel, and the Publisher channel.
Maximum number of entries in the log (50-500)	Number of entries in the log. The default value is 50.

E.6 Driver Image

Allows you to change the image associated with the driver. You can browse and select a different image from the default image.

The image associated with a driver is used by the Identity Manager Overview plug-in when showing the graphical representation of your Identity Manager configuration. Although storing an image is optional, it makes the overview display more intuitive.

NOTE: The driver image is maintained when a driver configuration is exported.

In iManager:

1. Click *Identity Manager* > *Identity Manager Overview*, then click *Search* to search for the driver set that is associated with the driver.
2. Browse to the driver, then click the upper right corner of the driver icon.
3. Click *Edit Properties* > *Driver Image*.

In Designer:

1. Open a project in the Modeler, then right-click the driver line and select *Properties* > *iManager Icon.*

E.7 Security Equals

Use the Security page to view or change the list of objects that the driver is explicitly security equivalent to. This object effectively has all rights of the listed objects.

If you add or delete an object in the list, the system automatically adds or deletes this object in that object's “Security Equal to Me” property. You don't need to add the [Public] trustee or the parent containers of this object to the list, because this object is already implicitly security equivalent to them.

In iManager:

1. Click *Identity Manager* > *Identity Manager Overview*, then click *Search* to search for the driver set that is associated with the driver.
2. Browse to the driver, then click the upper right corner of the driver icon.
3. Click *Edit Properties* > *Security Equals*.

Designer does not list the users the driver is security equivalent to.

E.8 Filter

Launches the Filter editor. You can edit the filter from this tab.

In iManager:

1. Click *Identity Manager* > *Identity Manager Overview*, then click *Search* to search for the driver set that is associated with the driver.
2. Browse to the driver, then click the upper right corner of the driver icon.
3. Click *Edit Properties* > *Filter*.

The Filter editor is accessed through the outline view in Designer.

1. In an open project, click the *Outline* tab.
2. Select the driver you want to manage the filter for, then click the plus sign to the left.
3. Double-click the *Filter* icon to launch the Filter Editor.

E.9 Edit Filter XML

Allows you to edit the filter directly in XML instead of using the Filter Editor.

In iManager:

1. Click *Identity Manager* > *Identity Manager Overview*, then click *Search* to search for the driver set that is associated with the driver.
2. Browse to the driver, then click the upper right corner of the driver icon.
3. Click *Edit Properties* > *Filter*.

You can edit the Filter in XML through the Filter Editor in Designer.

1. In an open project, click the *Outline* tab.
2. Select the driver you want to manage the filter for, then click the plus sign to the left.
3. Double-click the *Filter* icon to launch the Filter Editor, then click *XML Source* at the bottom of the Filter Editor.

E.10 Misc

Allows you to add a trace level to your driver. With the trace level set, DSTrace displays the Identity Manager events as the Metadirectory engine processes the events. The trace level only affects the driver it is set for. Use the trace level for troubleshooting issues with the driver when the driver is deployed. DSTrace displays the output of the specified trace level.

In iManager:

1. Click *Identity Manager > Identity Manager Overview*, then click *Search* to search for the driver set that is associated with the driver.
2. Browse to the driver, then click the upper right corner of the driver icon.
3. Click *Edit Properties > Misc*.

   See Table E-11 for a list of the driver trace levels.

In Designer:

1. Open a project in the Modeler, then right-click the driver line and select *Properties > Trace.*

   See Table E-11 for a list of the driver trace levels.

Table E-11 *Driver Trace Levels*

Option	Description
Trace level	Increases the amount of information displayed in DSTrace. Trace level 1 shows errors, but not the cause of the errors. If you want to see password synchronization information, set the trace level to 5.
Trace file	When a value is set in this field, all Java information for the driver is written to the file. The value for this field is the path for that file. As long as the file is specified, Java information is written to this file. If you do not need to debug Java, leave this field blank.
Trace file size limit	Allows you to set a limit for the Java trace file. If you set the file size to Unlimited, the file grows in size until there is no disk space left. **NOTE:** The trace file is created in multiple files. Identity Manager automatically divides the maximum file size by ten and creates ten separate files. The combined size of these files equals the maximum trace file size.
Trace name	Driver trace messages are prepended with the value entered in this field.
Use setting from Driver Set	This option is only available in Designer. It allows the driver to use the same setting that is set on the Driver Set object.

E.11 Excluded Objects

Use this page to create a list of objects that are not replicated to the application. Novell recommends that you add all objects that represent an administrative role to this list (for example, the Admin object).

In iManager:

1. Click *Identity Manager > Identity Manager Overview*, then click *Search* to search for the driver set that is associated with the driver.
2. Browse to the driver, then click the upper right corner of the driver icon.
3. Click *Edit Properties > Excluded Objects*.

Designer does not list the excluded objects.

E.12 Driver Manifest

The driver manifest is like a resumé for the driver. It states what the driver supports, and includes a few configuration settings. The driver manifest is created by default when the Driver object is imported. A network administrator usually does not need to edit the driver manifest.

In iManager:

1. Click *Identity Manager > Identity Manager Overview*, then click *Search* to search for the driver set that is associated with the driver.
2. Browse to the driver, then click the upper right corner of the driver icon.
3. Click *Edit Properties > Driver Manifest*.

In Designer:

1. Open a project in the Modeler, then right-click the driver line and select *Properties > Driver Manifest.*

E.13 Driver Inspector

The Driver Inspector page displays information about all of the objects associated with the driver.

- **Driver:** A link to run the *Driver Overview* on the driver that is being inspected.
- **Driver Set:** A link to run the *Driver Set Overview* of the driver set that holds the driver.
- **Delete:** Deletes the associations of the selected objects.
- **Refresh:** Select this option to re-read all of the objects associated with the driver and refresh the displayed information.
- **Actions:** Allows you to perform actions on the objects associated with the driver. Click *Actions* to expand the menu, which includes:
    - **Show All Associations:** Displays all objects associated with the driver.
    - **Filter for Disabled Associations:** Displays all objects associated with the driver that have a Disabled state.
    - **Filter for Manual Associations:** Displays all objects associated with the driver that have a Manual state.
    - **Filter for Migrate Associations:** Displays all objects associated with the driver that have a Migrate state.
    - **Filter for Pending Associations:** Displays all objects associated with the driver that have a Pending state.

    - **Filter for Processed Associations:** Displays all objects associated with the driver that have a Processed state.
    - **Filter for Undefined Associations:** Displays all objects associated with the driver that have an Undefined state.
    - **Association Summary:** Displays the state of all objects associated with the driver.
- **Object DN:** Displays the DN of the associated objects.
- **State:** Displays the association state of the object.
- **Object ID:** Displays the value of the association.

E.14 Driver Cache Inspector

The Driver Cache Inspector page uses a table format to display information about the cache file that stores events while the driver is stopped.

- **Driver:** A link to run the *Driver Overview* on the driver that is associated with this cache file.
- **Driver Set:** A link to run the *Driver Set Overview* on the driver set that holds the driver.
- **Driver's cache on:** Lists the server object that contains this instance of the cache file.
- **Start/Stop Driver icons:** Displays the current state of the driver and allows you to start or stop the driver.
- **Edit icon:** Allows you to edit the properties of the currently selected Server object.
- **Delete:** Deletes the selected items from the cache file.
- **Refresh:** Select this option to re-read the cache file and refresh the displayed information.
- **Show:** Limits the number of items to be displayed. The options are:
    - 25 per page
    - 50 per page
    - 100 per page
    - Other: Allows you to specify a desired number.
- **Actions:** Allows you to perform actions on the entries in the cache file. Click *Actions* to expand the menu, which includes:
    - **Expand All:** Expands all of the entries displayed in the cache file.
    - **Collapse All:** Collapses all of the entries displayed in the cache file.
    - **Go To:** Allows you to access a specified entry in the cache file. Specify the entry number, then click *OK*.
    - **Cache Summary:** Summarizes all of the events stored in the cache file.

E.15 Inspector

The Inspector displays information about the connected system without directly accessing the system. Designer does not have this option.

E.16 Server Variables

This page lets you enable and disable password synchronization and the associated options for the selected driver.

When setting up password synchronization, consider both the settings on this page for an individual driver and the Universal Password Configuration options in your password policies.

This page lets you control which password Identity Manager updates directly, either the Universal Password for an Identity Vault, or the Distribution Password used for password synchronization by Identity Manager.

However, Novell Modular Authentication Service (NMAS™) controls whether the various passwords inside the Identity Vault are synchronized with each other. Password Policies are enforced by NMAS, and they include settings for synchronizing Universal Password, NDS® Password, Distribution Password, and Simple Password.

To change these settings in iManager:

1 In iManager, select *Passwords > Password Policies*.

2 Select a password policy, then click *Edit*.

3 Select *Universal Password*.

This option is available from a drop-down list or a tab, depending on your version of iManager and your browser.

4 Select *Configuration Options*, make changes, then click *OK*.

NOTE: Enabling or disabling options on this page corresponds to values of True or False for certain global configuration values (GCVs) used for password synchronization in the driver parameters. Novell recommends that you edit them here in the graphical interface, instead of on the GCVs page. This interface helps ensure that you don't set conflicting values for the password synchronization GCVs.

Option	Description
Identity Manager accepts password (Publisher Channel)	If this option is enabled, Identity Manager allows passwords to flow from the connected system driver into the Identity Vault data store. Disabling this option means that no <*password*> elements are allowed to flow to Identity Manager. They are stripped out of the XML by a password synchronization policy on the Publisher channel. If this option is enabled, and the option below it for Distribution Password is disabled, a <*password*> value coming from the connected system is written directly to the Universal Password in the Identity Vault if it is enabled for the user. If the user's password policy does not enable Universal Password, the password is written to the NDS Password.

Option	Description
Use Distribution Password for password synchronization	To use this setting, you must have a version of eDirectory that supports Universal Password, regardless of whether you have enabled Universal Password in your password policies. If this option is enabled, a password value coming from the connected system is written to the Distribution Password. The Distribution Password is reversible, which means that it can be retrieved from the Identity Vault data store for password synchronization. It is used by Identity Manager for bidirectional password synchronization with connected systems. For Identity Manager to distribute passwords to connected systems, this option must be enabled. NMAS and Password policies control whether the Distribution Password is synchronized with other passwords in the Identity Vault. By default, the Distribution Password is the same as the Universal Password in the Identity Vault. If the password in the Identity Vault is to be independent of password synchronization, so that Identity Manager is a conduit only for synchronizing passwords among connected systems, change this default setting. In the Universal Password Configuration Options in a Password policy, disable *Synchronize Universal Password with Distribution Password*. This use of Identity Manager password synchronization is also referred to as “tunneling.”
Accept password only if it complies with user’s Password Policy	To use this setting, users must have a Password policy assigned that has Universal Password enabled, and Advanced Password Rules enabled and configured. If this option is chosen, Identity Manager does not write a password from this connected system to the Distribution Password in the Identity Manager data store or publish it to connected systems unless the password complies with the user's Password policy. By using the notification option that is also on this page, you can inform users when a password is not set because it is not compliant.

Option	Description
If password does not comply, ignore Password Policy on the connected system by resetting user’s password to the Distribution Password	This option lets you enforce Password policies on the connected system by replacing a password that does not comply. If you select this option, and a user's password on the connected system does not comply with the user's Password policy, Identity Manager resets the password on the connected system by using the Distribution Password from the Identity Vault data store. Keep in mind that if you do not select this option, user passwords can become out-of-sync on connected systems. By using the notification option that is also on this page, you can inform users when a password is not set or reset. Notification is especially helpful for this option. If the user changes to a password that is allowed by the connected system but rejected by Identity Manager because of the Password policy, the user won't know that the password has been reset until the user receives a notification or tries to log in to the connected system with the old password. **NOTE:** Consider the connected system's password policies when deciding whether to use this option. Some connected systems might not allow the reset because they don't allow you to repeat passwords.
Always accept password; ignore Password Policies	If you select this option, Identity Manager does not enforce the user's Password policy for this connected system. Identity Manager writes the password from this connected system to the Distribution Password in the Identity Vault data store, and distributes it to other connected systems, even if the password does not comply with the user's Password policy.
Application accepts passwords (Subscriber Channel)	If you select this option, the driver sends passwords from the Identity Vault data store to this connected system. This also means that if a user changes the password on a different connected system that is publishing passwords to the Distribution Password in the Identity Vault data store, the password is changed on this connected system. By default, the Distribution Password is the same as the Universal Password in the Identity Vault, so changes to the Universal Password made in the Identity Vault are also sent to the connected system. If you want the password in the Identity Vault to be independent of password synchronization, so that Identity Manager is a conduit only for synchronizing passwords among connected systems, you can change this default setting. In the Universal Password Configuration Options in a password policy, disable *Synchronize Universal Password with Distribution Password*. This use of password synchronization is also referred to as “tunneling.”
Notify the user of password synchronization failure via-email	If you select this option, e-mail is sent to the user if a password is not synchronized, set, or reset. The e-mail that is sent to the user is based on an e-mail template. This template is provided by the Password Synchronization application. However, for the template to work, you must customize it and specify an e-mail server to send the notification messages. **NOTE:** To set up e-mail notification, select *Passwords > Edit EMail Templates*.